7CCSMDM1 Data Mining Coursework 2

AnmolGuggari_K20105622

I. TEXT MINING

The total runtime is 4 seconds

1. In the first part of the text mining, we normalize the data, first : unique sentiments of the tweet.
    a. Positive-11422
    b. Negative-991
    c. 7Neutral-7713
    d. ExtremelyPositive-6624
    e. Extremely, Negative-5481

    Next step of normalization of the data, convert all the text into o lower case, replace non-alphabetical characters with whitespaces.

2. In this section, created 2 different data sets, first: tokenize the tweets (i.e., convert each into a list of words), count the total number of all words (including repetitions), the number of all distinct words and the 10 most frequent and in second part Remove stop words, words with ≤ 2 characters and recalculate the number of all words (including repetitions) and the 10 most frequent words in the modified corpus.

    Here in this data, we observe that, in first part post tokenization and when we count the total number of words it was 54852 and the top 10 frequent words were “the ,to, and , of, in , coronavirus ,19, covid, for, is "

    And post removing the stop word and the top 10 frequent words were: “coronavirus ,19, covid , prices , food , supermarket , store , grocery , people ,amp “

3. In what way this plot can be useful for deciding the size of the term document matrix? How many terms would you add in a term-document matrix for this data set?

    In the plot I have shown the histogram (line graph) in both increasing and decreasing order as well as used the log scale. All this helps to understand the pattern in the data. We can see that around 5 words appear in 40% of the documents. In the log plot we can see that after the first 2000 words everything else in the vocabulary appear in less than 1% of the documents. The final 10,000 words when arranged in the decreasing order appear in less than 0.0001 % of the documents. Based on these estimates I would be inclined to decrease the size of the term document matrix to keep only the first 10,000 terms when arranged in decreasing order. The final 10,000 words occur too infrequently to be able to helpful in training machine learning algorithms.

3. weighted avg, precision is 0.72 recall is 0.72, and f1-score is 0.72

II. IMAGE MINING

Before we begin, download all the library numpy, imageio, cv2, skimage. Have tried different library for each image to get good exposure with each library.

1. For avengers' image I have used the library pillow to open the image and to display the image size, later we convert it to black & white and then grayscale. The orginal image with size 630, 1200
    a. We can describe image as a function f where x belongs to [a,b] and y belongs to [c,d] which returns as output ranging between maximum and minimum pixel intensity values. - black and white image has more noise.
    b. Gray scale image - f: [a,b] * [c,d] -> [min,max](For gray-scale images, the output of the function is a range of possible values from the brightest pixel 255 to the darkest pixel 0) - gray scale image is much clearer, and less noise compared to black and white image.

2. Gaussian Noise is a statistical noise having a probability density function equal to normal distribution, also known as Gaussian Distribution. Random Gaussian function is added to Image function to generate this noise.
    a. Effect of Standard Deviation(sigma) on Gaussian noise: The magnitude of Gaussian Noise depends on the Standard Deviation(sigma). Noise Magnitude is directly proportional to the sigma value.
    b. I have used skimage library for this section and applied Gaussian mode and with variance 0.1 and latter applied uniform filter to the original image with size=(9, 9, 1)

3. K-mean segmentation: The main aim of the segmentation is to change the representation of an image that is more meaningful and easier to understand.
    a. Image segmentation is used to locate objects and boundaries(lines, curves, etc.) in images. Segmentation allows to label to every pixel in an image such that pixels with the same label share certain characteristics.
    b. I have used the library openCV and clustering with 5

4. Canny Edge- It finds the edges by looking for local maxima of the gradient off(x,y)
    a. The gradient is calculated using the derivative of a Gaussian filter.
    b. The method uses two thresholds to detect strong and weak edges and includes the weak edges in the output only if they are connected to strong edges. Therefore, this method is more likely to detect true weak edges.
    c. Probabilistic Hough - Hough transform is finding lines/curve in an image. Basically, applying some properties of Hough space, we can analyze and identify some groups of pixels that coordinates the properties such as being on a same line or crossing set of lines, Thus, providing us the information required to draw these lines.