Assignment – I

(Handling Unstructured Textual Data)

Q1: Based on Loading corpus

Download the IMDB movie review corpus from the following link:

https://drive.google.com/file/d/1DnRiVquqJ-IrUnBfO4i7ilnCO6mKHD_e/view?usp=sharing

(a) Load the dataset into a Pandas DataFrame
(b) Load first 100 reviews from the review column of DataFrame into a list of strings called as 'corpus'

Q2: Based on Pre-processing Corpus

Pre-process each document (i.e. each string of corpus list) so that all words are in lower case, there is no special symbols, url, numbers, or stopwords. Also, each word of the document is stemmed to its root word using Porter Stemmer

Q3: Based on Constructing Term Document Matrix

For the preprocessed corpus, construct a Term Document Matrix (using both inbuilt methods and from scratch; and the results of both should match), whose entries are:

(a) Binary
(b) Actual Term Frequency
(c) Term Frequency with length normalization
(d) Term Frequency-Inverse Document Frequency

Q4: Based on Co-occurrence Matrix

(a) Construct a term-term co-occurrence matrix from the pre-processed list of documents (obtained from Q2) whose each (ij)th entry is number of documents in which both ith and jth terms co-occur.
(b) Also, construct a Positive Pointwise Mutual Information (PPMI) whose (ij)th entry is computed as:

$$PPMI(i,j) = max\left(log\left(\frac{n(i,j)\ \mathrm{X|D|}}{n(i)\times n(j)}\right),0\right)$$

Where

$n(i,j)\ is\ number\ of\ documents\ in\ whch\ both\ i^{th}\ and\ j^{th}\ terms\ cooccur, n(i) is\ number\ of$

$documents\ in\ which\ i^{th}\ term\ occurs, n(j)\ is\ number\ of\ documents\ in\ which\ j^{th}\ term\ occurs$

where |D| represents total number of documents.