Assignment:-4

Question1:-

Preprocessing: I used os library to walk in entire directories and make an all directory's documents list and then one by one I took all the documents and open the documents one by one. By using nltk word_tokenizer I tokenize and by using num2Word I change all the numbers in the text to words and I used Stopwotds to the removal and remove all the punctuations from the text content. Then the entire document content refined and then I made an inverted index on the bases of tf and store this dictionary. From this dictionary and idf dictionary I made all the document vectors dictionary as docId it's key of every vector for fast query processing. And store this dictionary as a pickle file.

Methodology: After preprocessing I took the dictionary and take a query to preprocess this query similar to the above documents and make a query vector now I took dot product from this normalized query vector and save the similarity score in the dictionary now I sort this dictionary by its value and print top k documents.

Question2:-

Preprocessing:- Same as above. Only methodology changes.

Methodology:-

For the (a) part simply above solution works and it takes a query and shows the retrieved ranked documents.

For part (b) I opt to choose the document manually as mentioned in the comment section. by vrutti ma'am Because making p static means reducing the user freedom. But if you need to implement the concept of p% then this is implemented in the section (e) separately because we have given ground truth there.

For part (c) After performing relevant feedback, we printed all the marked documents which come in retrieved documents. Because there is no point if I print the same documents I took input from the user that's why I printed the marked relevant documents in the retrieved documents. But I also printed the * relevant document marked by the user so I printed both.

For part(d). Simply I implemented the Roccio's algorithm.

For part(e). For this part, the mechanism is same as above but I restricted the program with the given query and their given ground truth and I took p as 10% of given k value. I am taking 4 feedback iterations every time and each iteration plots pr curve and also a map for every refined query point and in the last TSNE plots with 2 components.

For part(f). for the justification, my initial query vector shifts towards the relevant documents and we got a new shifted query vector in each iteration. After appending all the queries in each iteration I used TSNE and made these high dimensional query vectors as 2 dimensional and plot those queries by using matplotlib package.

Justification: Each time we are getting different shifted 2 dimension coordinate for each refined query because our query is shifting towards the relevant documents that's why we are getting a new query every time.