READ ME

Note: I have provided the pickled file for Inverted list and document vectors which will be required for quicker execution of the code.

Please find the links below:

Inverted List Pickle file :

https://drive.google.com/open?id=1VASouad70sEhF2TX27RUFUKUb4ITkFoT

Document Vectors Pickle file:

https://drive.google.com/open?id=1PhzDb0zzUpzKnkrNPfpE365EoG5olwf9

20newsgroup data

https://drive.google.com/open?id=1b0xBiqshlued82Fo99uo9_NoPFUi1f63

How to Run

1. Get the files from the links given above and edit the path in your code accordingly.

   All the python cells in the code are well commented with first comment in the code as the cell ID.

   ```
   ### CELL Inverted List ### → change the 20newsgroup data set path
   ## CELL 5 to execute ## → change Inverted list pickle file path
   ## CELL 6 to execute ## → change the 20newsgroup data set path
   ## CELL 8 to execute ## → change Document vectors Pickle file path
   ## CELL 14 to execute ## → change the 20newsgroup data set path
   ```

2. run the cells sequentially in the order given below. Please note that I have intentionally asked you to avoid executing few code cells since some produced output is already pickled.

```
## CELL 1 to execute ## → importing NLTK
## CELL 2 to execute ## → installing num2words
## CELL 3 to execute ## → import all necessities
## CELL 4 to execute ## → download stopwords from NLTK
### CELL Inverted List ### → Avoid executing
### Code to Create Pickle file for Inverted List ###--> Avoid
executing
## CELL 5 to execute ## → extract inverted List from Pickled file
## CELL 6 to execute ## → calculate total number of documents in the
corpus
## CELL 7 to execute ## → Final inverted list (Inv) with TF-IDF
values
## Code to create document vectors (Docs_vec) for all 5k documents
in the corpus ### → Avoid executing
## Code to dump all the document vectors created in the pickle file
called "Docs_Vec_IR4.pickle"--> Avoid executing
## CELL 8 to execute ## → load all the document vectors from
"Docs_Vec_IR4.pickle
## CELL 9 to execute ## → Cosine(Q_vec,k)
## CELL 10 to execute ## → graph() function is used to plot 2D TSNE
## CELL 11 to execute ## → calc_PR() is used to plot precision -
recall curve
## CELL 12 to execute ## → calculate Average precision for the set
of queries
## CELL 13 to execute ## → calculate mean average precision MAP
## CELL 14 to execute ## → Heart string of this code file
```

Functions Description

1. **def cosine(Q_vec,k):**

   This function takes 2 paramers Q_vec ::: Query Vector and k ::: Number of results user wants to display.This function returns Ranked top k documents with highest cosine similarity with the query vector.

2. **def graph(R_vecs,NR_vecs,Q_vec)**

   graph() function is used to plot 2D TSNE graph of the vectors to demonstrate known relevant , non relevan and the query vector.

3. **def calc_PR(rel_array):**
   Calc_PR() is used to plot precision recall curve for each iteration of relevance feedback

4. **def AP(rel_array):** Function to calculate Average precision for the set of queries

4. **def MAP(res_arr):** Function to caculate mean average precision MAP.