ASSIGNMENT 2
NAME:-Anand Sharma (MT19059)

Analysis Report

Question 1:-

Total files read:-467
Total Unique tokens formed after preprocesiing :-50907
Total document and token matrix size is :-411645

TOP 10 documents retrieved :-using log normalization in TF score and inverse document frequency in idf score

```
['disco', 'can', 'be', 'fun']
RETRIEVED DOCUMENTS BASAED ON JACCARD COEFFICINT
['disco.be.fun', 'discocanbefun.txt', 'alissadl.txt', 'quarter.c15', 'hareleph.txt',
'snowmaid.txt', 'cameloto.hum', 'startrek.txt', 'brain.damage', 's&m_that']
467
50907
411645
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE with TITLE considered same weight
['fgoose.txt', 'disco.be.fun', 'discocanbefun.txt', 'cybersla.txt', 'chik', 'archive',
'hitch2.txt', 'cooldark.sto', 'cooldark.txt', 'hitch3.txt']
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE WITH TITLE given more weightage
['disco.be.fun', 'discocanbefun.txt', '100west.txt', '13chil.txt', '14.lws', '16.lws', '17.lws',
'18.lws', '19.lws', '20.lws']
50907
411643
RETRIEVED DOCUMENTS BASAED ON COSINE SIMILARITY
['disco.be.fun', 'discocanbefun.txt', 'chik', 'fgoose.txt', 'quarter.c15', 'alissadl.txt',
'snowmaid.txt', 'hareleph.txt', 'fear.hum', 'brain.damage']
```

TOP 10 documents retrieved :-using log normalization in TF score and inverse document frequency in idf score

```
['50000', 'variety', 'of', 'flowers']
RETRIEVED DOCUMENTS BASAED ON JACCARD COEFFICINT
['ghost', 'wall.art', 'lgoldbrd.txt', 'day.in.mcdonald', 'mcdonaldl.txt', 'ccm.txt', 'fantas.hum',
'tin', 'bgcspoof.txt', 'bulolli2.txt']
467
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE with TITLE considered same weight
['timem.hac', 'breaks1.asc', 'hitch3.txt', 'ghost', 'outcast.dos', 'cybersla.txt', 'bgcspoof.txt',
'gulliver.txt', 'radar_ra.txt', 'fic4']
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE WITH TITLE given more weightage
['ghost', '100west.txt', '13chil.txt', '14.lws', '16.lws', '17.lws', '18.lws', '19.lws', '20.lws',
'3gables.txt']
50907
411643
RETRIEVED DOCUMENTS BASAED ON COSINE SIMILARITY
['fic4', 'ghost', 'tin', 'wall.art', 'lgoldbrd.txt', 'stainles.ana', 'bram', 'bulolli2.txt',
'ccm.txt', 'day.in.mcdonald']
```

TOP 10 documents retrieved :-using double normalization in TF score and inverse document frequency smooth in idf score

```
['without', 'the', 'drive', 'of', 'rebeccah', 's', 'insistence', 'kate', 'lost', 'her',
'momentum', 'she', 'stood', 'next', 'a', 'slatted', 'oak', 'bench', 'canisters', 'still',
'clutched', 'surveying']
RETRIEVED DOCUMENTS BASAED ON JACCARD COEFFICINT
['ghost', 'quarter.c6', 'foxnstrk.txt', 'quarter.c4', 'quarter.c15', 'narciss.txt',
'graymare.txt', 'quarter.c9', 'redragon.txt', 'quarter.c17']
```

```
467
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE with TITLE considered same weight
['ghost', 'vgilante.txt', 'radar_ra.txt', 'sre_finl.txt', 'enc', 'hellmach.txt', 'cooldark.sto',
'cooldark.txt', 'outcast.dos', 'dakota.txt']
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE WITH TITLE given more weightage
['100west.txt', '13chil.txt', '14.lws', '16.lws', '17.lws', '18.lws', '19.lws', '20.lws',
'3gables.txt', '3lpigs.txt']
```

TOP 10 documents retrieved :-using raw TF score and inverse document frequency smooth in idf score

```
['without', 'the', 'drive', 'of', 'rebeccah', 's', 'insistence', 'kate', 'lost', 'her',
'momentum', 'she', 'stood', 'next', 'a', 'slatted', 'oak', 'bench', 'canisters', 'still',
'clutched', 'surveying']
RETRIEVED DOCUMENTS BASAED ON JACCARD COEFFICINT
['ghost', 'quarter.c6', 'foxnstrk.txt', 'quarter.c4', 'quarter.c15', 'narciss.txt',
'graymare.txt', 'quarter.c9', 'redragon.txt', 'quarter.c17']
467
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE with TITLE considered same weight
['ghost', 'vgilante.txt', 'radar_ra.txt', 'sre_finl.txt', 'enc', 'hellmach.txt', 'cooldark.sto',
'cooldark.txt', 'outcast.dos', 'dakota.txt']
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE WITH TITLE given more weightage
['100west.txt', '13chil.txt', '14.lws', '16.lws', '17.lws', '18.lws', '19.lws', '20.lws',
'3gables.txt', '3lpigs.txt']
50907
411643
RETRIEVED DOCUMENTS BASAED ON COSINE SIMILARITY
['ghost', 'enc', 'vday.hum', 'fic7', 'sre_finl.txt', 'bulolli1.txt', 'cameloto.hum',
'graymare.txt', 'gold3ber.txt', 'vgilante.txt']
```

TOP 10 documents retrieved :-using raw TF score and inverse document frequency in idf score

```
['disco', 'can', 'be', 'fun']
RETRIEVED DOCUMENTS BASAED ON JACCARD COEFFICINT
['disco.be.fun', 'discocanbefun.txt', 'alissadl.txt', 'quarter.c15', 'hareleph.txt',
'snowmaid.txt', 'cameloto.hum', 'startrek.txt', 'brain.damage', 's&m_that']
467
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE with TITLE considered same weight
['cybersla.txt', 'archive', 'fgoose.txt', 'disco.be.fun', 'discocanbefun.txt', 'hitch2.txt',
'cooldark.sto', 'cooldark.txt', 'hitch3.txt', 'brain.damage']
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE WITH TITLE given more weightage
['disco.be.fun', 'discocanbefun.txt', '100west.txt', '13chil.txt', '14.lws', '16.lws', '17.lws',
'18.lws', '19.lws', '20.lws']
50907
411643
RETRIEVED DOCUMENTS BASAED ON COSINE SIMILARITY
['disco.be.fun', 'discocanbefun.txt', 'fgoose.txt', 'chik', 'quarter.c15', 'startrek.txt',
'brain.damage', 'fear.hum', 'keeping.insanit', 'snowmaid.txt']
```

TOP 10 documents retrieved :-using raw TF score and inverse document frequency in idf score

```
['the', 'adventure', 'of', 'the', 'adventure']
RETRIEVED DOCUMENTS BASAED ON JACCARD COEFFICINT
['holmesbk.txt', 'imagin.hum', 'advtthum.txt', 'szechuan', 'testpilo.hum', 'lure.txt', 's&m_that',
'panama.txt', '7voysinb.txt', 'plescopm.txt']
467
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE with TITLE considered same weight
['holmesbk.txt', 'gulliver.txt', '7voysinb.txt', 'hound-b.txt', 'archive', 'empty.txt',
'lure.txt', 'rocket.sf', '3student.txt', 'bruce-p.txt']
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE WITH TITLE given more weightage
['empty.txt', '3student.txt', 'bruce-p.txt', '3gables.txt', '6napolen.txt', 'enginer.txt',
'solitary.txt', 'wisteria.txt', 'abbey.txt', 'goldenp.txt']
50907
411643
RETRIEVED DOCUMENTS BASAED ON COSINE SIMILARITY
['holmesbk.txt', '7voysinb.txt', 'testpilo.hum', 'imagin.hum', 'panama.txt', 'empty.txt',
'plescopm.txt', 'lure.txt', 'advtthum.txt', 'szechuan']
```

TOP 10 documents retrieved :-using raw TF score and inverse document frequency in idf score

```
Enter the query to be searched"the adventure of the adventure --- "
Enter the value of no of documents to be retrieved10
adventure
RETRIEVED DOCUMENTS BASAED ON JACCARD COEFFICINT
['the-tree.txt', 'disco.be.fun', 'discocanbefun.txt', 'berternie.txt', 'how.ernie.bert',
'bagel.man', 'bagelman.txt', 'foxncrow.txt', 'bestwish', 'deal']
467
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE with TITLE considered same weight
['vgilante.txt', 'dakota.txt', 'sick-kid.txt', 'archive', 'batlslau.txt', 'cybersla.txt',
'beggars.txt', 'robotech', 'sre06.txt', 'hitch2.txt']
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE WITH TITLE given more weightage
['vgilante.txt', 'sre_sei.txt', 'hitch2.txt', 'cybersla.txt', 'archive', 'sre06.txt',
'history5.txt', 'cooldark.sto', 'cooldark.txt', 'friends.txt']
51060
454647
RETRIEVED DOCUMENTS BASAED ON COSINE SIMILARITY
['berternie.txt', 'how.ernie.bert', 'disco.be.fun', 'discocanbefun.txt', 'bagel.man',
'bagelman.txt', 'modemhippy.txt', 'sre06.txt', 'bookem.1', 'bulolli2.txt']


Enter the query to be searchedgable animals animals
Enter the value of no of documents to be retrieved10
gable animals animals
['gable', 'animal', 'animal']
RETRIEVED DOCUMENTS BASAED ON JACCARD COEFFICINT
['monkking.txt', 'hotline4.txt', 'redragon.txt', 'horsdonk.txt', 'dwar', 'bran', 'clevdonk.txt',
'ccm.txt', 'quarter.c15', 'toilet.s']
467
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE with TITLE considered same weight
['gulliver.txt', 'wisteria.txt', 'lionmane.txt', 'cybersla.txt', 'dakota.txt', 'aesop11.txt',
'hitch2.txt', 'timem.hac', '3gables.txt', 'outcast.dos']
RETRIEVED DOCUMENTS BASAED ON TF IDF SCORE WITH TITLE given more weightage
['3gables.txt', '100west.txt', '13chil.txt', '14.lws', '16.lws', '17.lws', '18.lws', '19.lws',
'20.lws', '3lpigs.txt']
45098
432121
RETRIEVED DOCUMENTS BASAED ON COSINE SIMILARITY
['lionmane.txt', 'monkking.txt', '3gables.txt', 'wisteria.txt', 'redragon.txt', 'hotline4.txt',
'horsdonk.txt', 'bran', 'clevdonk.txt', 'dwar']
45124
45124
432121
RETRIEVED DOCUMENTS BASED ON COSINE SIMILARITY WITH TITLE GIVEN MORE WEIGHT
['3gables.txt', 'lionmane.txt', 'wisteria.txt', 'monkking.txt', 'hotline4.txt', 'redragon.txt',
'horsdonk.txt', 'bran', 'dwar', 'ccm.txt']
```

Analysis:-

1. when we give more weightage to title we can see in above cases the file with title will be ranked above others without title . It is shown in above shots
2. when we use different tf score and idf score calculation we are getting set of documents that are naerly same some files are different
3. I get most accurate files in log normalization and Inverse frequency score
4. Some files ranking vary when we use different idf norm and raw TF score . Files ranking changes and some are different

5. With double normalization i am getting different result . Some are common but new files are also retrieved
6. when we give empty query no files are retrieved
7. Based on 3 methods Cosine Similarity has performed best and given more accurate documents retrieved then jaccard . Cosine then tf-idf retrieval then jaccard
8. Normalized TF is best and normalized IDF is best in accurate result

PRO and CONS

Jaccard coefficient:-

1. less time complexity
2. easier implementation of code
3. it doesnot use order and frequency of terms in ranking the document
4. not an efficient approach for Document retrieval as terms frequency are ignored

TF-IDF BASED RETRIEVAl:-

1. used terms frequency and document terms for ranking therefore more accurate result
2. Title weightage is used to show the importance of terms in title and ranked above others
3. Used various techniques for calculation of TF and IDF and do the normalization for more accurate result
4. Good ranking algorithm compared to Jaccard
5. Ordering of words are not considered
6. not special attention to title in basic TF idf model

COSINE SIMILARITY:-

1. value are between 0 and 1 so ranking is more accurate and precised

2. used to calculate more accurately then other models as it takes dot product and normalize it
3. Better than TF IDF
4. more memory is required than previous models
5. Time complexity is more then others models
6. Memory is wasted

Question 2:-

(1) Query:-
```
Enter the query anand sharma Information Retrieval assignment question
enter the value of k 5
['anand', 'sharma', 'information', 'retrieval', 'assignment', 'question']
```

Suggestions:-
```
['anaconda', 'gangland', 'mainland', 'amanda', 'anacondas']
['sahara', 'harm', 'sarcoma', 'sham', 'sherman']
information
retrieval
assignment
question
```

(2)Query:-
```
Enter the query anand sharma
enter the value of k 5
['anand', 'sharma']
```
Suggestions:_
```
['anaconda', 'gangland', 'mainland', 'amanda', 'anacondas']
['sahara', 'harm', 'sarcoma', 'sham', 'sherman']
```

(3) Query:-
```
Enter the query sherlock hlmoes fats and furiosu travellre
enter the value of k 5
['sherlock', 'hlmoes', 'fats', 'and', 'furiosu', 'travellre']
```

suggestions :-
```
sherlock
['haloes', 'holmes', 'helmets', 'hoes', 'almoners']
fats
and
['furious', 'curios', 'furies', 'furiosity', 'furiously']
['travelled', 'traveller', 'travellers', 'ravelled', 'traverse']
```

In case user give incorrect words as query our system will give suggestions based on it accurately