INFORMATION RETRIEVAL ASSIGNMENT 1

SUBMITTED BY:-

ANURAG GAUTAM (MT22015) **MUSKAAN GUPTA(MT22113)**
SALONI GARG(MT22063)

Ques1 Data Preprocessing

i) Relevant Text Extraction

Ans1i) Methodologies

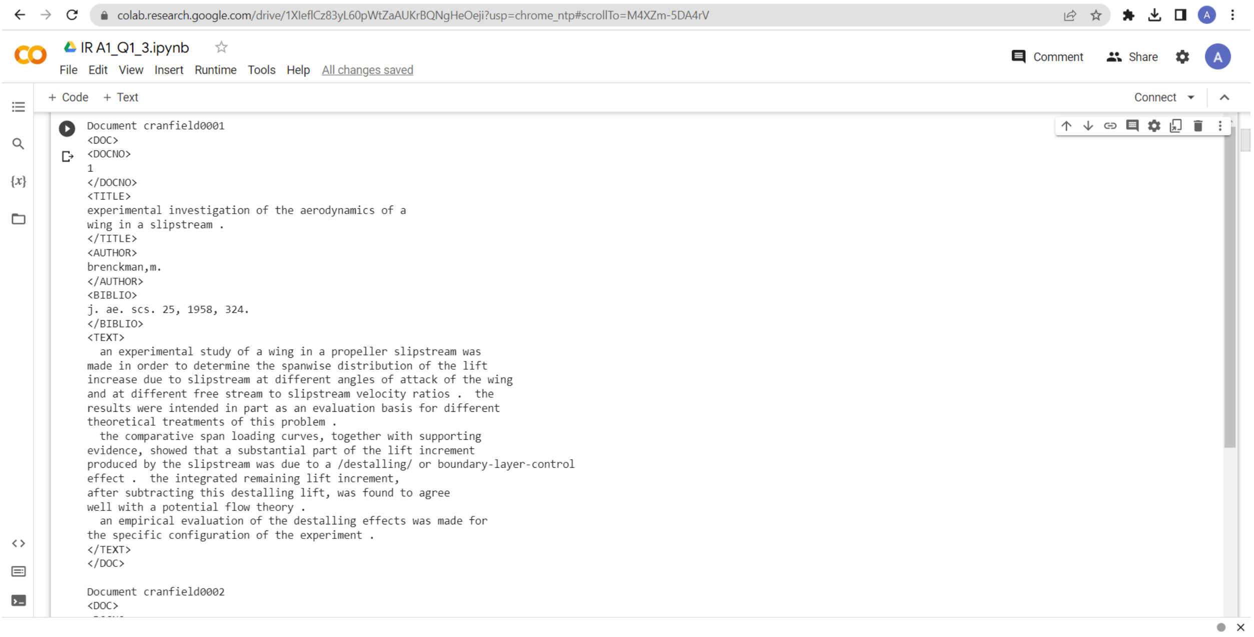
Using the BeautifulSoup library in Python, we have performed preprocessing on HTML data by extracting the content from the <title> and <text> tags. Afterward, we have utilized file-handling techniques in Python to write the extracted content into a single file.

Assumptions

We assume we have all files with <title> and <text>

Results

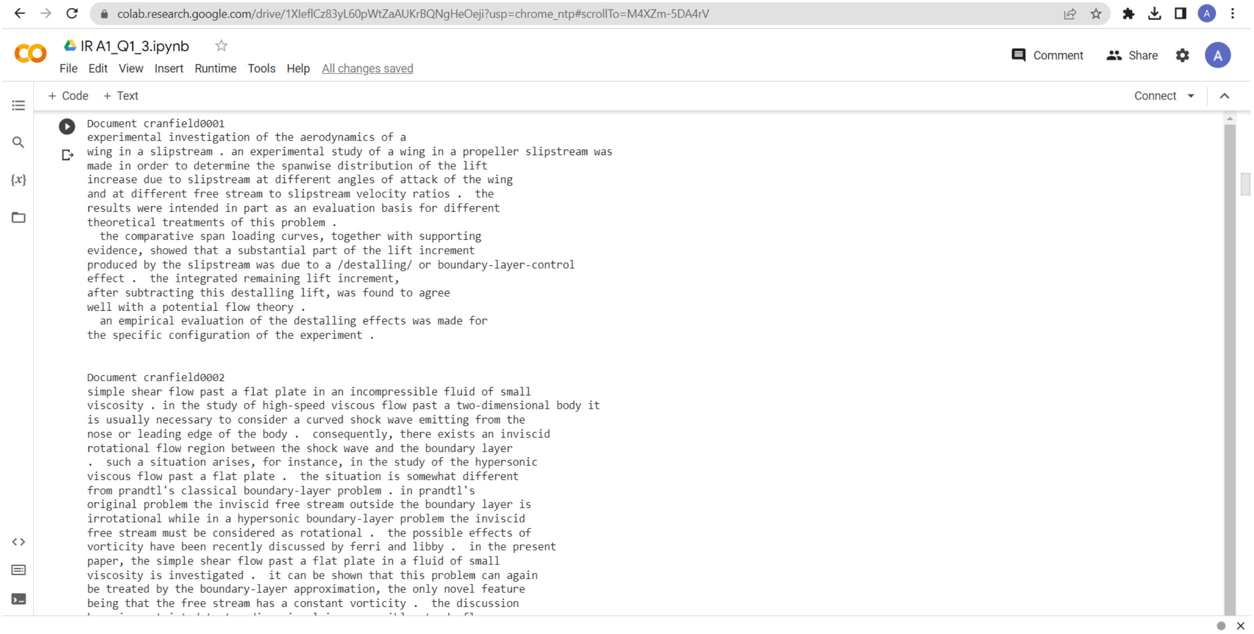
Before Preprocessing


colab.research.google.com/drive/1XleflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=M4XZm-5DA4rV
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
Document cranfield0001
<DOC>
<DOCNO>
1
</DOCNO>
<TITLE>
experimental investigation of the aerodynamics of a
wing in a slipstream .
</TITLE>
<AUTHOR>
brenckman,m.
</AUTHOR>
<BIBLIO>
j. ae. scs. 25, 1958, 324.
</BIBLIO>
<TEXT>
  an experimental study of a wing in a propeller slipstream was
made in order to determine the spanwise distribution of the lift
increase due to slipstream at different angles of attack of the wing
and at different free stream to slipstream velocity ratios .  the
results were intended in part as an evaluation basis for different
theoretical treatments of this problem .
  the comparative span loading curves, together with supporting
evidence, showed that a substantial part of the lift increment
produced by the slipstream was due to a /destalling/ or boundary-layer-control
effect .  the integrated remaining lift increment,
after subtracting this destalling lift, was found to agree
well with a potential flow theory .
  an empirical evaluation of the destalling effects was made for
the specific configuration of the experiment .
</TEXT>
</DOC>

Document cranfield0002
<DOC>
```

After preprocessing


colab.research.google.com/drive/1XlefICz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=M4XZm-5DA4rV
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
Document cranfield0001
experimental investigation of the aerodynamics of a
wing in a slipstream . an experimental study of a wing in a propeller slipstream was
made in order to determine the spanwise distribution of the lift
increase due to slipstream at different angles of attack of the wing
and at different free stream to slipstream velocity ratios .  the
results were intended in part as an evaluation basis for different
theoretical treatments of this problem .
  the comparative span loading curves, together with supporting
evidence, showed that a substantial part of the lift increment
produced by the slipstream was due to a /destalling/ or boundary-layer-control
effect .  the integrated remaining lift increment,
after subtracting this destalling lift, was found to agree
well with a potential flow theory .
  an empirical evaluation of the destalling effects was made for
the specific configuration of the experiment .


Document cranfield0002
simple shear flow past a flat plate in an incompressible fluid of small
viscosity . in the study of high-speed viscous flow past a two-dimensional body it
is usually necessary to consider a curved shock wave emitting from the
nose or leading edge of the body .  consequently, there exists an inviscid
rotational flow region between the shock wave and the boundary layer
.  such a situation arises, for instance, in the study of the hypersonic
viscous flow past a flat plate .  the situation is somewhat different
from prandtl's classical boundary-layer problem . in prandtl's
original problem the inviscid free stream outside the boundary layer is
irrotational while in a hypersonic boundary-layer problem the inviscid
free stream must be considered as rotational .  the possible effects of
vorticity have been recently discussed by ferri and libby .  in the present
paper, the simple shear flow past a flat plate in a fluid of small
viscosity is investigated .  it can be shown that this problem can again
be treated by the boundary-layer approximation, the only novel feature
being that the free stream has a constant vorticity .  the discussion
```

ii) Preprocessing

Ans1ii) Methodologies

A custom data preprocessing function was created, which was used to generate a dictionary of filename indexes that contained all the tokens created after the data had been preprocessed for a particular file. The function performs the following operations on the dataset:

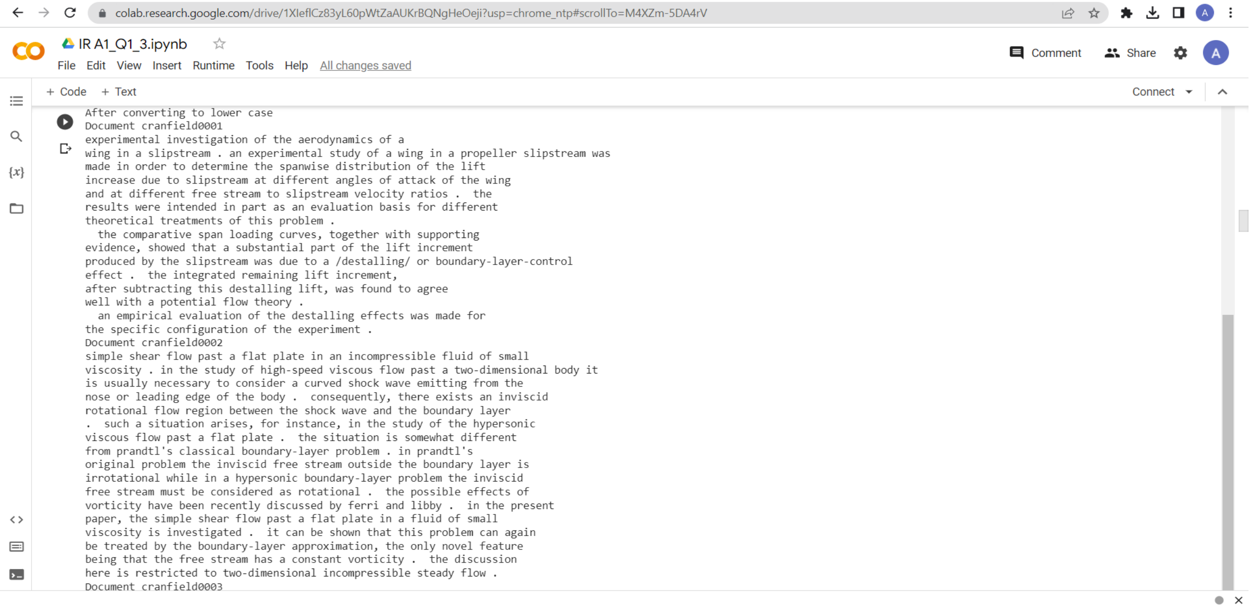
- Lowercase the text
- Perform tokenization
- Remove stopwords
- Remove punctuation
- Remove blank space tokens

Assumptions

Nothing assumed

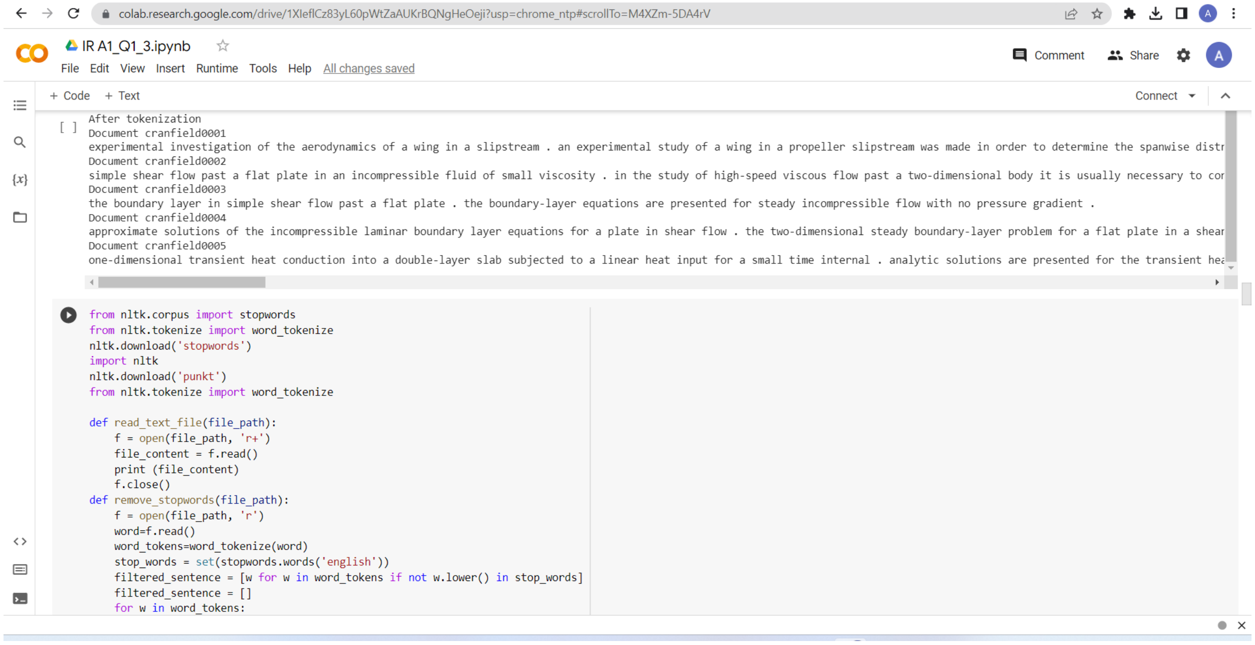
Results

After lowercasing:


colab.research.google.com/drive/1XleflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=M4XZm-5DA4rV
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
After converting to lower case
Document cranfield0001
experimental investigation of the aerodynamics of a
wing in a slipstream . an experimental study of a wing in a propeller slipstream was
made in order to determine the spanwise distribution of the lift
increase due to slipstream at different angles of attack of the wing
and at different free stream to slipstream velocity ratios .  the
results were intended in part as an evaluation basis for different
theoretical treatments of this problem .
  the comparative span loading curves, together with supporting
evidence, showed that a substantial part of the lift increment
produced by the slipstream was due to a /destalling/ or boundary-layer-control
effect .  the integrated remaining lift increment,
after subtracting this destalling lift, was found to agree
well with a potential flow theory .
  an empirical evaluation of the destalling effects was made for
the specific configuration of the experiment .
Document cranfield0002
simple shear flow past a flat plate in an incompressible fluid of small
viscosity . in the study of high-speed viscous flow past a two-dimensional body it
is usually necessary to consider a curved shock wave emitting from the
nose or leading edge of the body .  consequently, there exists an inviscid
rotational flow region between the shock wave and the boundary layer
.  such a situation arises, for instance, in the study of the hypersonic
viscous flow past a flat plate .  the situation is somewhat different
from prandtl's classical boundary-layer problem . in prandtl's
original problem the inviscid free stream outside the boundary layer is
irrotational while in a hypersonic boundary-layer problem the inviscid
free stream must be considered as rotational .  the possible effects of
vorticity have been recently discussed by ferri and libby .  in the present
paper, the simple shear flow past a flat plate in a fluid of small
viscosity is investigated .  it can be shown that this problem can again
be treated by the boundary-layer approximation, the only novel feature
being that the free stream has a constant vorticity .  the discussion
here is restricted to two-dimensional incompressible steady flow .
Document cranfield0003
```

After tokenization:

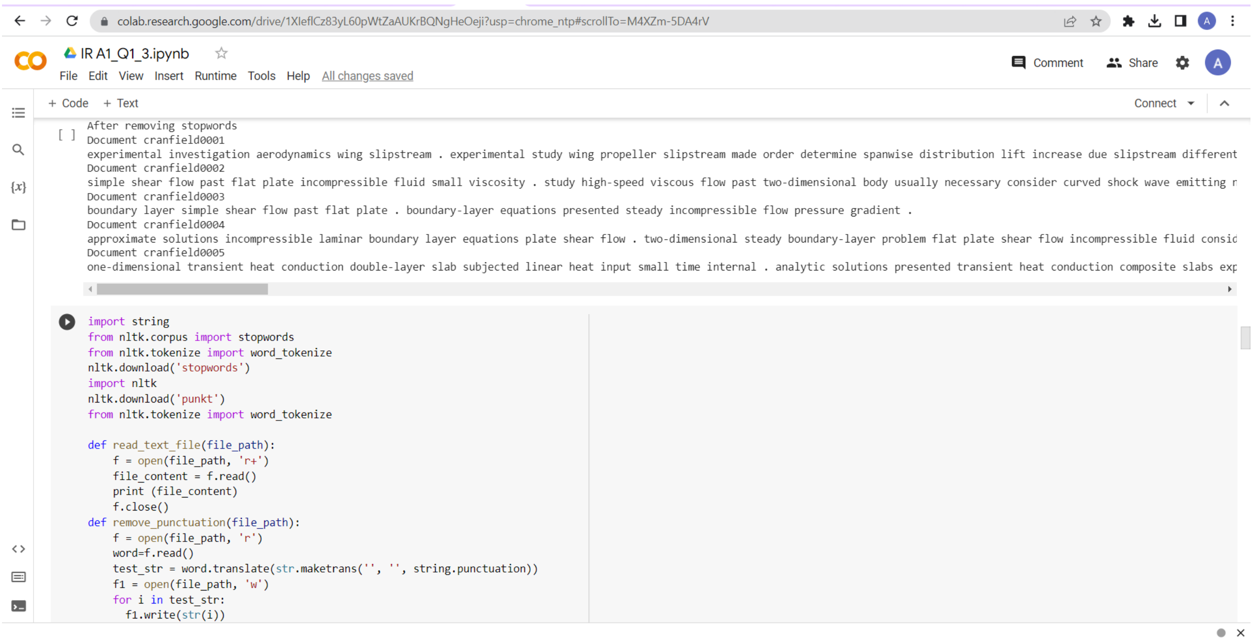

colab.research.google.com/drive/1XleflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=M4XZm-5DA4rV
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
After tokenization
Document cranfield0001
experimental investigation of the aerodynamics of a wing in a slipstream . an experimental study of a wing in a propeller slipstream was made in order to determine the spanwise distr
Document cranfield0002
simple shear flow past a flat plate in an incompressible fluid of small viscosity . in the study of high-speed viscous flow past a two-dimensional body it is usually necessary to cor
Document cranfield0003
the boundary layer in simple shear flow past a flat plate . the boundary-layer equations are presented for steady incompressible flow with no pressure gradient .
Document cranfield0004
approximate solutions of the incompressible laminar boundary layer equations for a plate in shear flow . the two-dimensional steady boundary-layer problem for a flat plate in a shear
Document cranfield0005
one-dimensional transient heat conduction into a double-layer slab subjected to a linear heat input for a small time internal . analytic solutions are presented for the transient hea
```

```python
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
nltk.download('stopwords')
import nltk
nltk.download('punkt')
from nltk.tokenize import word_tokenize

def read_text_file(file_path):
    f = open(file_path, 'r+')
    file_content = f.read()
    print (file_content)
    f.close()
def remove_stopwords(file_path):
    f = open(file_path, 'r')
    word=f.read()
    word_tokens=word_tokenize(word)
    stop_words = set(stopwords.words('english'))
    filtered_sentence = [w for w in word_tokens if not w.lower() in stop_words]
    filtered_sentence = []
    for w in word_tokens:
```

After removing stopwords

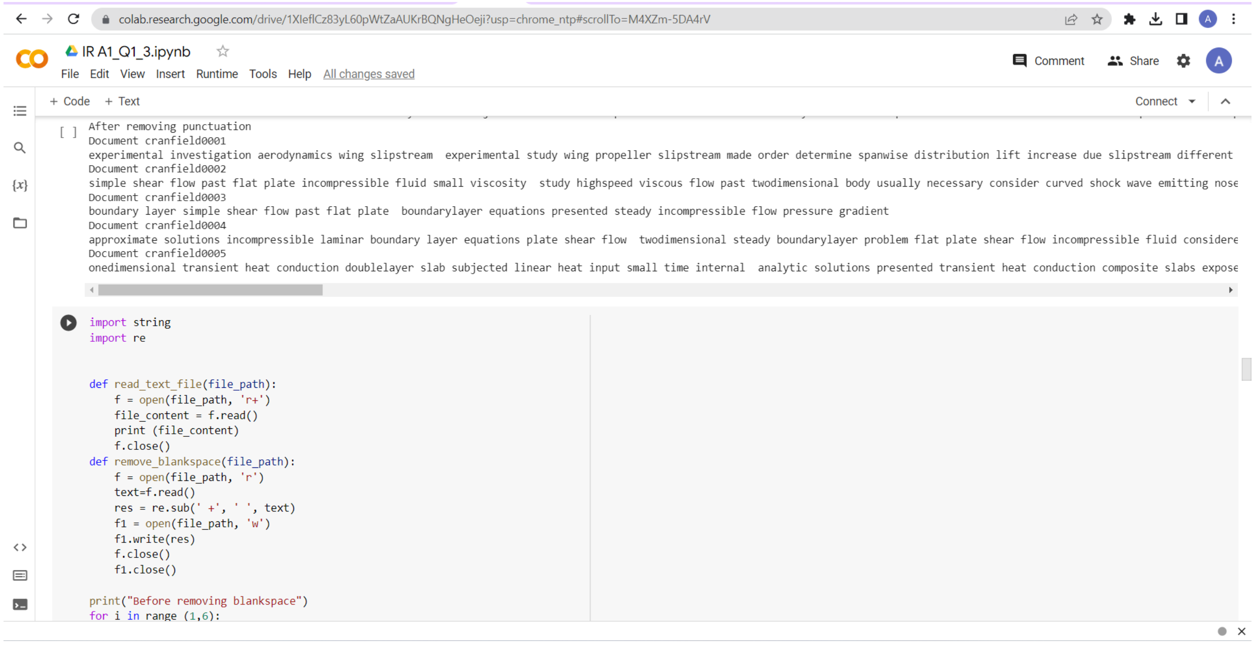

colab.research.google.com/drive/1XIeflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=M4XZm-5DA4rV
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
After removing stopwords
Document cranfield0001
experimental investigation aerodynamics wing slipstream . experimental study wing propeller slipstream made order determine spanwise distribution lift increase due slipstream different
Document cranfield0002
simple shear flow past flat plate incompressible fluid small viscosity . study high-speed viscous flow past two-dimensional body usually necessary consider curved shock wave emitting r
Document cranfield0003
boundary layer simple shear flow past flat plate . boundary-layer equations presented steady incompressible flow pressure gradient .
Document cranfield0004
approximate solutions incompressible laminar boundary layer equations plate shear flow . two-dimensional steady boundary-layer problem flat plate shear flow incompressible fluid consic
Document cranfield0005
one-dimensional transient heat conduction double-layer slab subjected linear heat input small time internal . analytic solutions presented transient heat conduction composite slabs exp
```

```python
import string
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
nltk.download('stopwords')
import nltk
nltk.download('punkt')
from nltk.tokenize import word_tokenize

def read_text_file(file_path):
    f = open(file_path, 'r+')
    file_content = f.read()
    print (file_content)
    f.close()
def remove_punctuation(file_path):
    f = open(file_path, 'r')
    word=f.read()
    test_str = word.translate(str.maketrans('', '', string.punctuation))
    f1 = open(file_path, 'w')
    for i in test_str:
      f1.write(str(i))
```

After removing punctuation


colab.research.google.com/drive/1XIeflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=M4XZm-5DA4rV
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

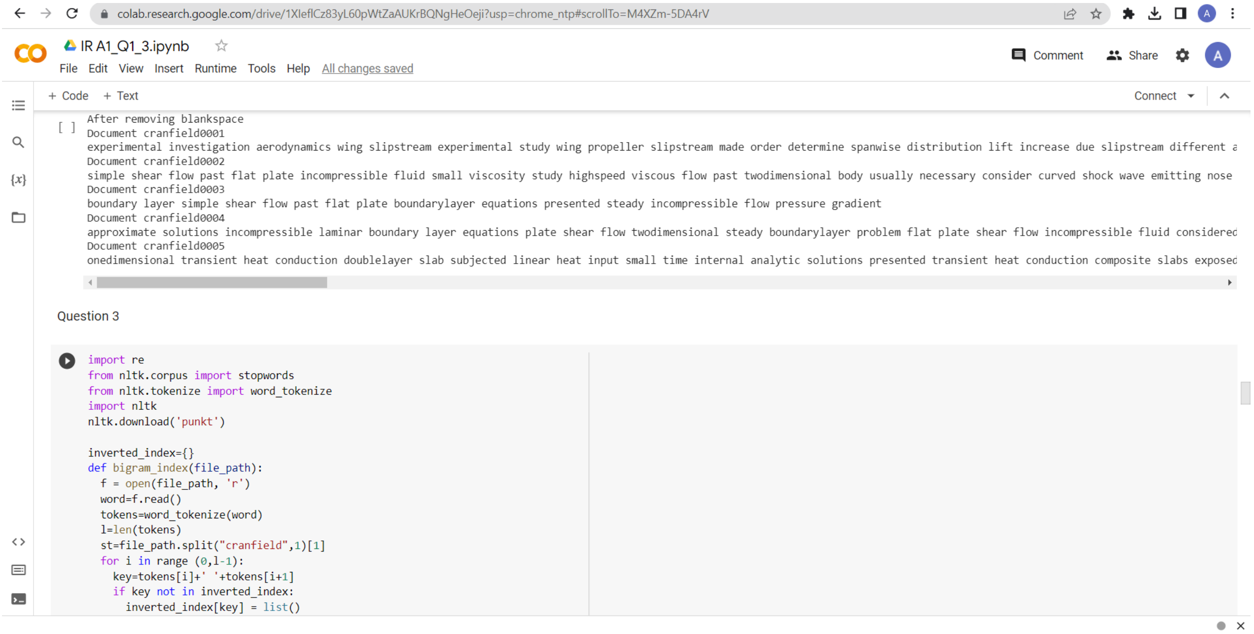
```
After removing punctuation
Document cranfield0001
experimental investigation aerodynamics wing slipstream  experimental study wing propeller slipstream made order determine spanwise distribution lift increase due slipstream different
Document cranfield0002
simple shear flow past flat plate incompressible fluid small viscosity  study highspeed viscous flow past twodimensional body usually necessary consider curved shock wave emitting nose
Document cranfield0003
boundary layer simple shear flow past flat plate  boundarylayer equations presented steady incompressible flow pressure gradient
Document cranfield0004
approximate solutions incompressible laminar boundary layer equations plate shear flow  twodimensional steady boundarylayer problem flat plate shear flow incompressible fluid considere
Document cranfield0005
onedimensional transient heat conduction doublelayer slab subjected linear heat input small time internal  analytic solutions presented transient heat conduction composite slabs expose
```

```python
import string
import re


def read_text_file(file_path):
    f = open(file_path, 'r+')
    file_content = f.read()
    print (file_content)
    f.close()
def remove_blankspace(file_path):
    f = open(file_path, 'r')
    text=f.read()
    res = re.sub(' +', ' ', text)
    f1 = open(file_path, 'w')
    f1.write(res)
    f.close()
    f1.close()

print("Before removing blankspace")
for i in range (1,6):
```

After removing blank spaces



colab.research.google.com/drive/1XleflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=M4XZm-5DA4rV
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
After removing blankspace
Document cranfield0001
experimental investigation aerodynamics wing slipstream experimental study wing propeller slipstream made order determine spanwise distribution lift increase due slipstream different a
Document cranfield0002
simple shear flow past flat plate incompressible fluid small viscosity study highspeed viscous flow past twodimensional body usually necessary consider curved shock wave emitting nose
Document cranfield0003
boundary layer simple shear flow past flat plate boundarylayer equations presented steady incompressible flow pressure gradient
Document cranfield0004
approximate solutions incompressible laminar boundary layer equations plate shear flow twodimensional steady boundarylayer problem flat plate shear flow incompressible fluid considered
Document cranfield0005
onedimensional transient heat conduction doublelayer slab subjected linear heat input small time internal analytic solutions presented transient heat conduction composite slabs exposed
```

Question 3

```
import re
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
import nltk
nltk.download('punkt')

inverted_index={}
def bigram_index(file_path):
  f = open(file_path, 'r')
  word=f.read()
  tokens=word_tokenize(word)
  l=len(tokens)
  st=file_path.split("cranfield",1)[1]
  for i in range (0,l-1):
    key=tokens[i]+' '+tokens[i+1]
    if key not in inverted_index:
      inverted_index[key] = list()
```

Ques2 Boolean Queries

Ans2 Methodologies

In this task, we have utilized the output datasets from the previous questions to construct a unigram inverted index through a posting list. To begin, we have created a dataset dictionary that consists of individual words, followed by the creation of a posting list dictionary, as shown in the result.

Next, we have given a prompt for the user to input the number of queries and the phrase, The phrase is then preprocessed using essential techniques to form query tokens. These tokens will be utilized in various custom functions we have created, including OR, AND, NOT, ORNOT, and ANDNOT, to evaluate the query token list through the inverted index.

In addition, we have used the pickle module to save and load the unigram inverted index.

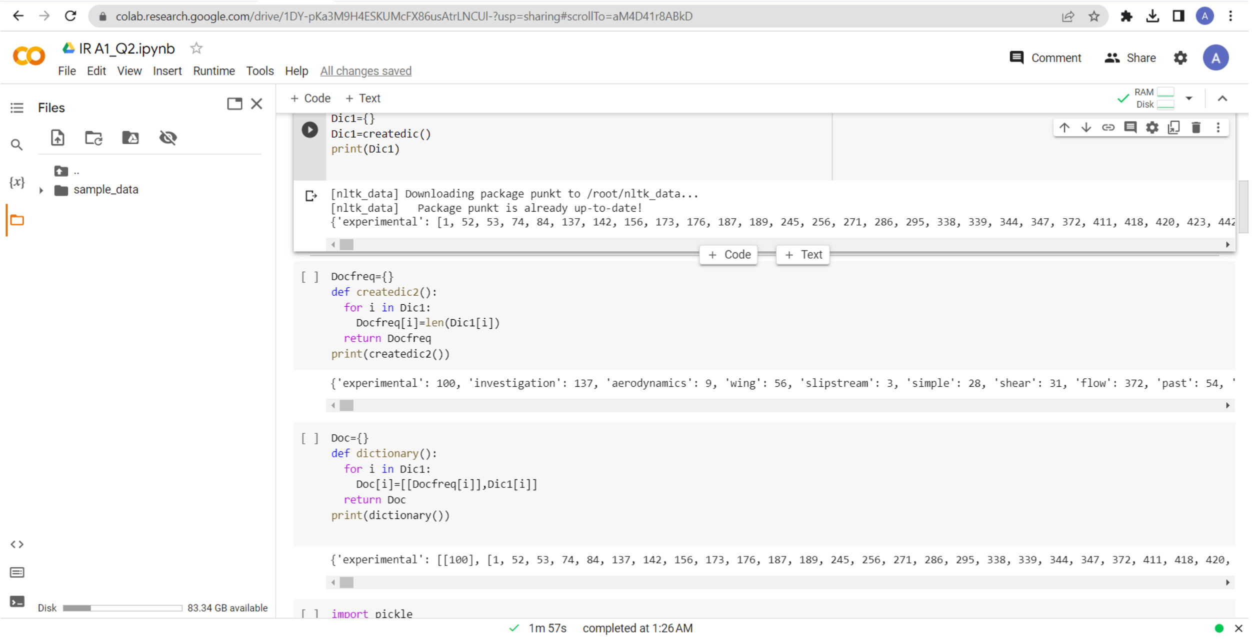
Assumption

We have assumed that the user will enter a phrase and the boolean operations separately. So, then we preprocess them both separately.

Results

The below figure shows the implementation of the unigram inverted index. We have first made a dictionary that contains the list for every word indicating in what documents the corresponding to that key. Then count the document frequency for each unigram and store it in that key's list.

Inverted index for unigram


colab.research.google.com/drive/1DY-pKa3M9H4ESKUMcFX86usAtrLNCUI-?usp=sharing#scrollTo=aM4D41r8ABkD
IR A1_Q2.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Files
sample_data
+ Code + Text
RAM
Disk
Disk 83.34 GB available
1m 57s completed at 1:26 AM

```
Dic1={}
Dic1=createdic()
print(Dic1)
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
{'experimental': [1, 52, 53, 74, 84, 137, 142, 156, 173, 176, 187, 189, 245, 256, 271, 286, 295, 338, 339, 344, 347, 372, 411, 418, 420, 423, 442
```

```
Docfreq={}
def createdic2():
  for i in Dic1:
    Docfreq[i]=len(Dic1[i])
  return Docfreq
print(createdic2())
```

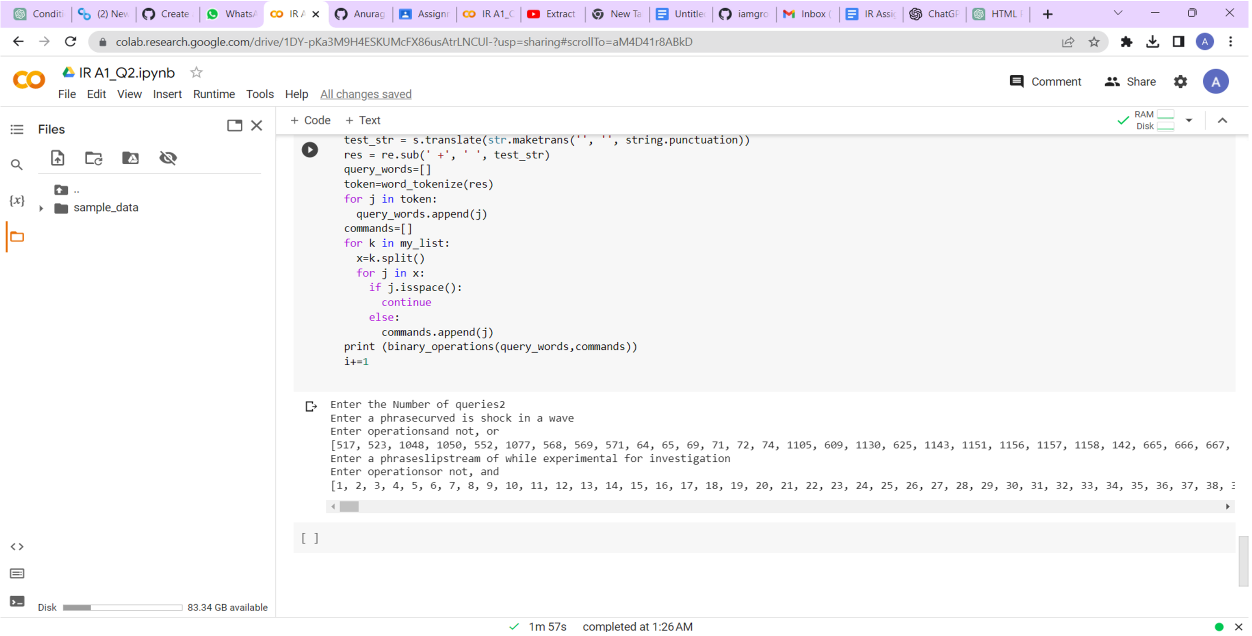
```
{'experimental': 100, 'investigation': 137, 'aerodynamics': 9, 'wing': 56, 'slipstream': 3, 'simple': 28, 'shear': 31, 'flow': 372, 'past': 54, '
```

```
Doc={}
def dictionary():
  for i in Dic1:
    Doc[i]=[[Docfreq[i]],Dic1[i]]
  return Doc
print(dictionary())
```

```
{'experimental': [[100], [1, 52, 53, 74, 84, 137, 142, 156, 173, 176, 187, 189, 245, 256, 271, 286, 295, 338, 339, 344, 347, 372, 411, 418, 420,
```

```
import pickle
```

Output for the inputted queries by the user


colab.research.google.com/drive/1DY-pKa3M9H4ESKUMcFX86usAtrLNCUI-?usp=sharing#scrollTo=aM4D41r8ABkD
IR A1_Q2.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Files
sample_data
+ Code + Text
RAM
Disk
Disk 83.34 GB available
1m 57s completed at 1:26 AM

```
  test_str = s.translate(str.maketrans('', '', string.punctuation))
  res = re.sub(' +', ' ', test_str)
  query_words=[]
  token=word_tokenize(res)
  for j in token:
    query_words.append(j)
  commands=[]
  for k in my_list:
    x=k.split()
    for j in x:
      if j.isspace():
        continue
      else:
        commands.append(j)
  print (binary_operations(query_words,commands))
  i+=1
```

```
Enter the Number of queries2
Enter a phrasecurved is shock in a wave
Enter operationsand not, or
[517, 523, 1048, 1050, 552, 1077, 568, 569, 571, 64, 65, 69, 71, 72, 74, 1105, 609, 1130, 625, 1143, 1151, 1156, 1157, 1158, 142, 665, 666, 667,
Enter a phraseslipstream of while experimental for investigation
Enter operationsor not, and
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 3
```

Ques3 Phrase Queries

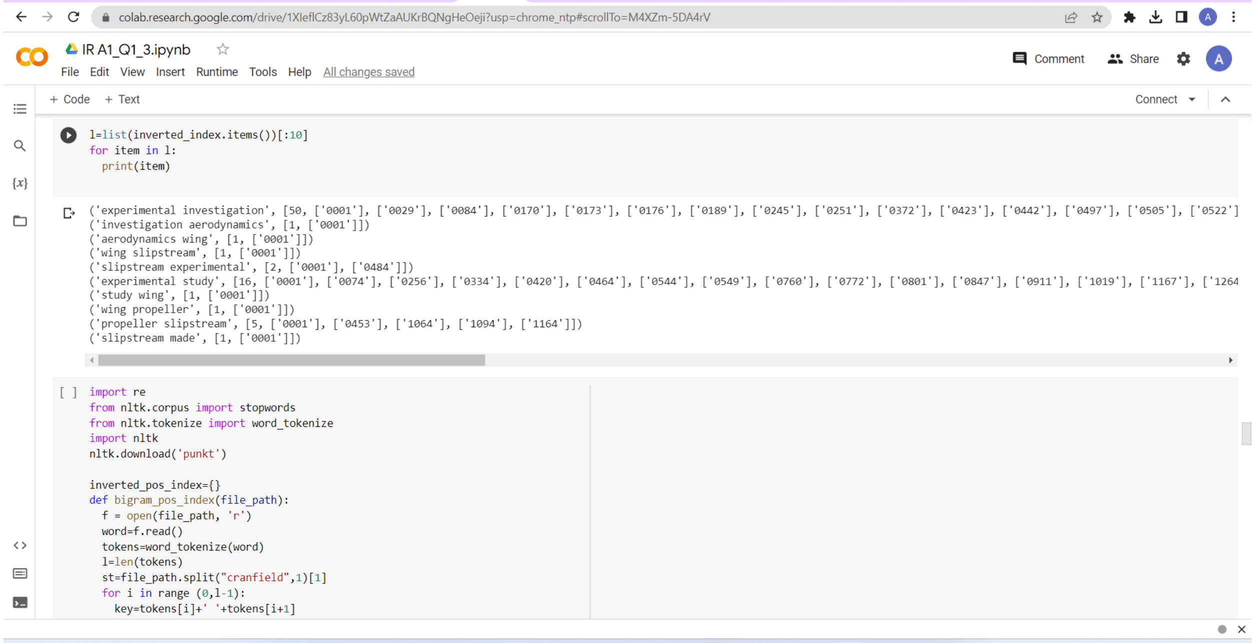
Ans3 Methodologies

In this task, we have utilized the output datasets from the 1st question to construct a bigram inverted index through a posting list. To begin, we have created a dataset dictionary that consists of bigram words, followed by the creation of a posting list dictionary, as shown in the result.

Similarly, we have created a dictionary for the bigram positional index.

Result:

Inverted index for bigrams


colab.research.google.com/drive/1XleflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=M4XZm-5DA4rV
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

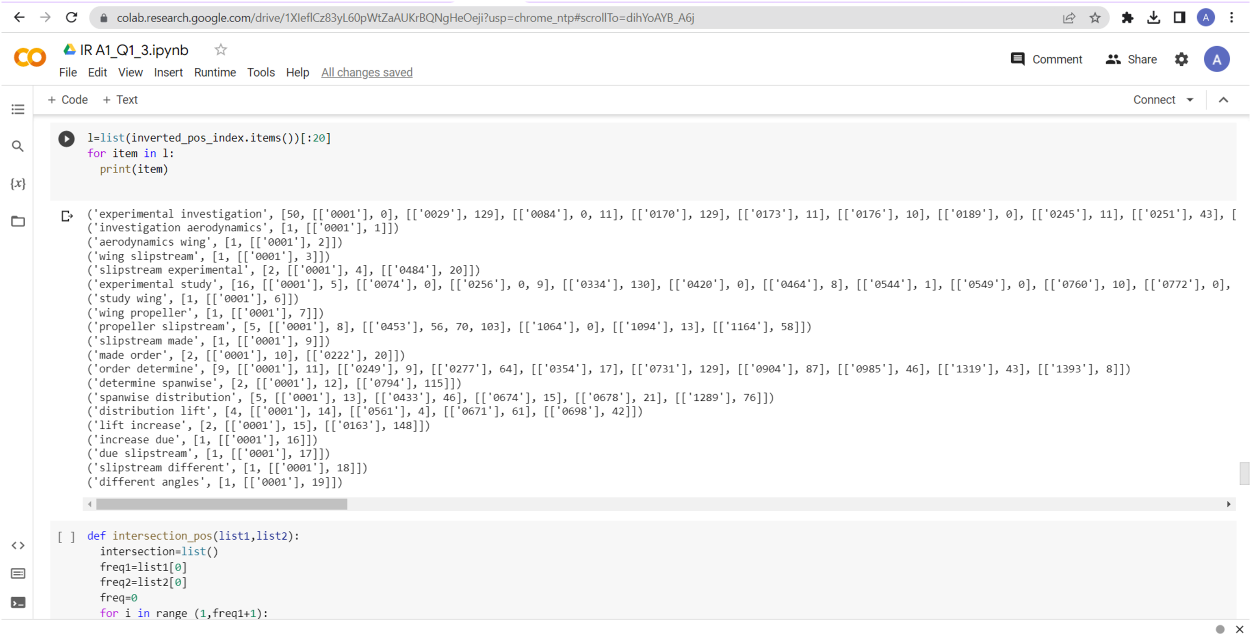
```
l=list(inverted_index.items())[:10]
for item in l:
  print(item)
```

```
('experimental investigation', [50, ['0001'], ['0029'], ['0084'], ['0170'], ['0173'], ['0176'], ['0189'], ['0245'], ['0251'], ['0372'], ['0423'], ['0442'], ['0497'], ['0505'], ['0522']
('investigation aerodynamics', [1, ['0001']])
('aerodynamics wing', [1, ['0001']])
('wing slipstream', [1, ['0001']])
('slipstream experimental', [2, ['0001'], ['0484']])
('experimental study', [16, ['0001'], ['0074'], ['0256'], ['0334'], ['0420'], ['0464'], ['0544'], ['0549'], ['0760'], ['0772'], ['0801'], ['0847'], ['0911'], ['1019'], ['1167'], ['1264
('study wing', [1, ['0001']])
('wing propeller', [1, ['0001']])
('propeller slipstream', [5, ['0001'], ['0453'], ['1064'], ['1094'], ['1164']])
('slipstream made', [1, ['0001']])
```

```
[ ] import re
    from nltk.corpus import stopwords
    from nltk.tokenize import word_tokenize
    import nltk
    nltk.download('punkt')

    inverted_pos_index={}
    def bigram_pos_index(file_path):
      f = open(file_path, 'r')
      word=f.read()
      tokens=word_tokenize(word)
      l=len(tokens)
      st=file_path.split("cranfield",1)[1]
      for i in range (0,l-1):
        key=tokens[i]+' '+tokens[i+1]
```

Positional Bigram Inverted Index

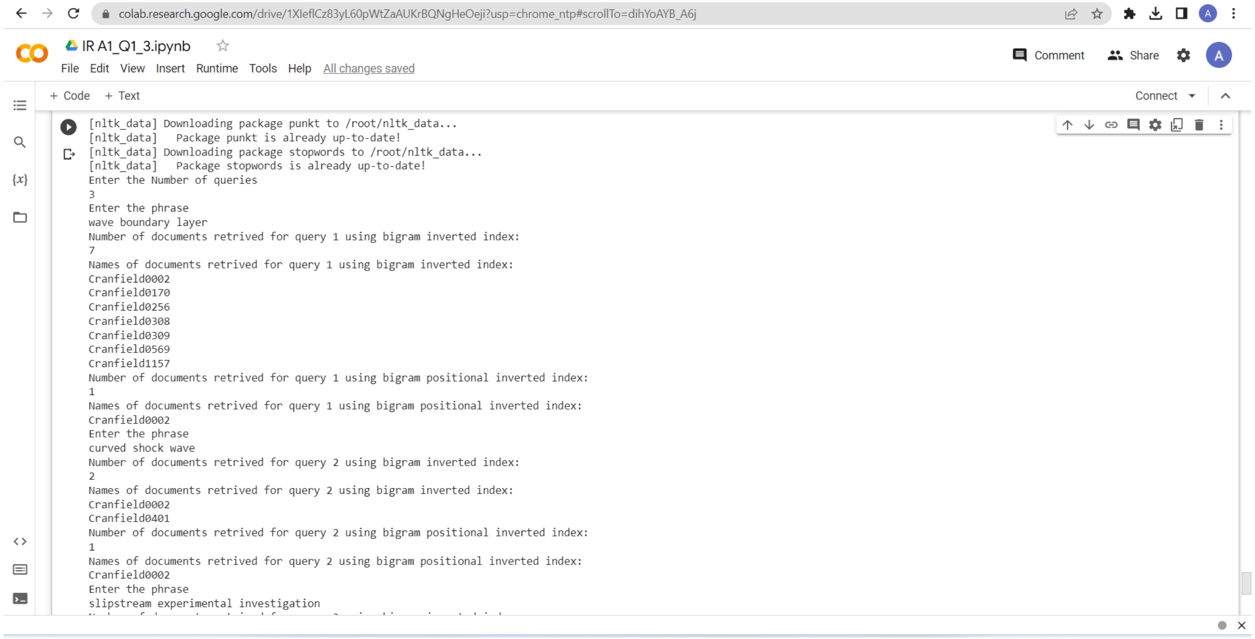

colab.research.google.com/drive/1XleflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=dihYoAYB_A6j
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
l=list(inverted_pos_index.items())[:20]
for item in l:
  print(item)
```

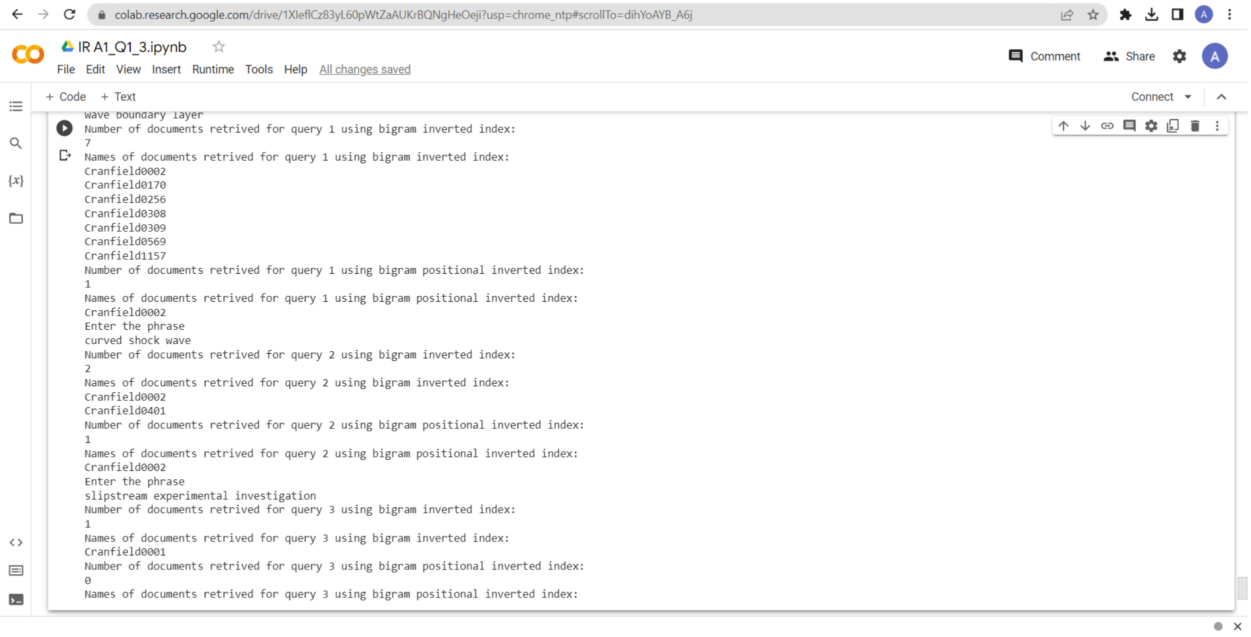
```
('experimental investigation', [50, [['0001'], 0], [['0029'], 129], [['0084'], 0, 11], [['0170'], 129], [['0173'], 11], [['0176'], 10], [['0189'], 0], [['0245'], 11], [['0251'], 43], [
('investigation aerodynamics', [1, [['0001'], 1]])
('aerodynamics wing', [1, [['0001'], 2]])
('wing slipstream', [1, [['0001'], 3]])
('slipstream experimental', [2, [['0001'], 4], [['0484'], 20]])
('experimental study', [16, [['0001'], 5], [['0074'], 0], [['0256'], 0, 9], [['0334'], 130], [['0420'], 0], [['0464'], 8], [['0544'], 1], [['0549'], 0], [['0760'], 10], [['0772'], 0],
('study wing', [1, [['0001'], 6]])
('wing propeller', [1, [['0001'], 7]])
('propeller slipstream', [5, [['0001'], 8], [['0453'], 56, 70, 103], [['1064'], 0], [['1094'], 13], [['1164'], 58]])
('slipstream made', [1, [['0001'], 9]])
('made order', [2, [['0001'], 10], [['0222'], 20]])
('order determine', [9, [['0001'], 11], [['0249'], 9], [['0277'], 64], [['0354'], 17], [['0731'], 129], [['0904'], 87], [['0985'], 46], [['1319'], 43], [['1393'], 8]])
('determine spanwise', [2, [['0001'], 12], [['0794'], 115]])
('spanwise distribution', [5, [['0001'], 13], [['0433'], 46], [['0674'], 15], [['0678'], 21], [['1289'], 76]])
('distribution lift', [4, [['0001'], 14], [['0561'], 4], [['0671'], 61], [['0698'], 42]])
('lift increase', [2, [['0001'], 15], [['0163'], 148]])
('increase due', [1, [['0001'], 16]])
('due slipstream', [1, [['0001'], 17]])
('slipstream different', [1, [['0001'], 18]])
('different angles', [1, [['0001'], 19]])
```

```
[ ] def intersection_pos(list1,list2):
      intersection=list()
      freq1=list1[0]
      freq2=list2[0]
      freq=0
      for i in range (1,freq1+1):
```

Output For Phrase Queries


colab.research.google.com/drive/1XleflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=dihYoAYB_A6j
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
Enter the Number of queries
3
Enter the phrase
wave boundary layer
Number of documents retrived for query 1 using bigram inverted index:
7
Names of documents retrived for query 1 using bigram inverted index:
Cranfield0002
Cranfield0170
Cranfield0256
Cranfield0308
Cranfield0309
Cranfield0569
Cranfield1157
Number of documents retrived for query 1 using bigram positional inverted index:
1
Names of documents retrived for query 1 using bigram positional inverted index:
Cranfield0002
Enter the phrase
curved shock wave
Number of documents retrived for query 2 using bigram inverted index:
2
Names of documents retrived for query 2 using bigram inverted index:
Cranfield0002
Cranfield0401
Number of documents retrived for query 2 using bigram positional inverted index:
1
Names of documents retrived for query 2 using bigram positional inverted index:
Cranfield0002
Enter the phrase
slipstream experimental investigation
```


colab.research.google.com/drive/1XleflCz83yL60pWtZaAUKrBQNgHeOeji?usp=chrome_ntp#scrollTo=dihYoAYB_A6j
IR A1_Q1_3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect

```
wave boundary layer
Number of documents retrived for query 1 using bigram inverted index:
7
Names of documents retrived for query 1 using bigram inverted index:
Cranfield0002
Cranfield0170
Cranfield0256
Cranfield0308
Cranfield0309
Cranfield0569
Cranfield1157
Number of documents retrived for query 1 using bigram positional inverted index:
1
Names of documents retrived for query 1 using bigram positional inverted index:
Cranfield0002
Enter the phrase
curved shock wave
Number of documents retrived for query 2 using bigram inverted index:
2
Names of documents retrived for query 2 using bigram inverted index:
Cranfield0002
Cranfield0401
Number of documents retrived for query 2 using bigram positional inverted index:
1
Names of documents retrived for query 2 using bigram positional inverted index:
Cranfield0002
Enter the phrase
slipstream experimental investigation
Number of documents retrived for query 3 using bigram inverted index:
1
Names of documents retrived for query 3 using bigram inverted index:
Cranfield0001
Number of documents retrived for query 3 using bigram positional inverted index:
0
Names of documents retrived for query 3 using bigram positional inverted index:
```