Information Retrieval

Assignment-2

Shubham Agrawal (MT22124)   Amrita Aash (MT22011)   Ayush Agarwal (MT22095)

Question 1

Methodology:

1. TF-IDF Matrix

- We extracted the text between TITLE and TEXT tags using the following regular expressions:

  ◆ `((?<=\<TITLE\>)(.|\n)*(?=\<\/TITLE\>))`

  ◆ `((?<=\<TEXT\>)(.|\n)*(?=\<\/TEXT\>))`

- Then we concatenated the texts, overwrote them on the corresponding file, and showed the file's contents before and after the concatenation step.

- Then for each of the pre-processing steps, we created a function to show the contents of the file before & after pre-processing.

- All the tokens were combined for each document to form the word corpus.

- Nested dictionaries were computed for the five weighting schemes for Term Frequency computation. Each term, if present in the document, is a key in the outer dictionary, and for each such key, the value is also a dictionary where the key is the document id, and its corresponding value is the term frequency. If any term is missing from the outer dictionary, it indicates that the term is not present in any of the 400 docs. Similarly, if any document id is missing from the inner dictionary, the document doesn't contain that term.

- Inverted index was picked from the previous assignment and was used to compute the Inverse Document Frequency for each term using a dictionary data structure.
- Five TF-IDF matrices were formed for computing the TF-IDF score for each document and each term in the word corpus using the five different term frequency nested dictionaries. These matrices were represented as data frames where each row is the document id, and each column is a term.

- After the user gives a query, we preprocess it, compute the TF of the query using each of the specified weighting schemes, computed the tfidf vector for the query, found the cosine similarity of the query with each of the documents (from the above TF-IDF matrices) for respective TF computation and stored it in a data frame, sorted them in descending order of Cosine similarity scores and printed the top 5 doc ids along with their scores. The IDF used for computing the TF-IDF of the query was the same as found in the document.

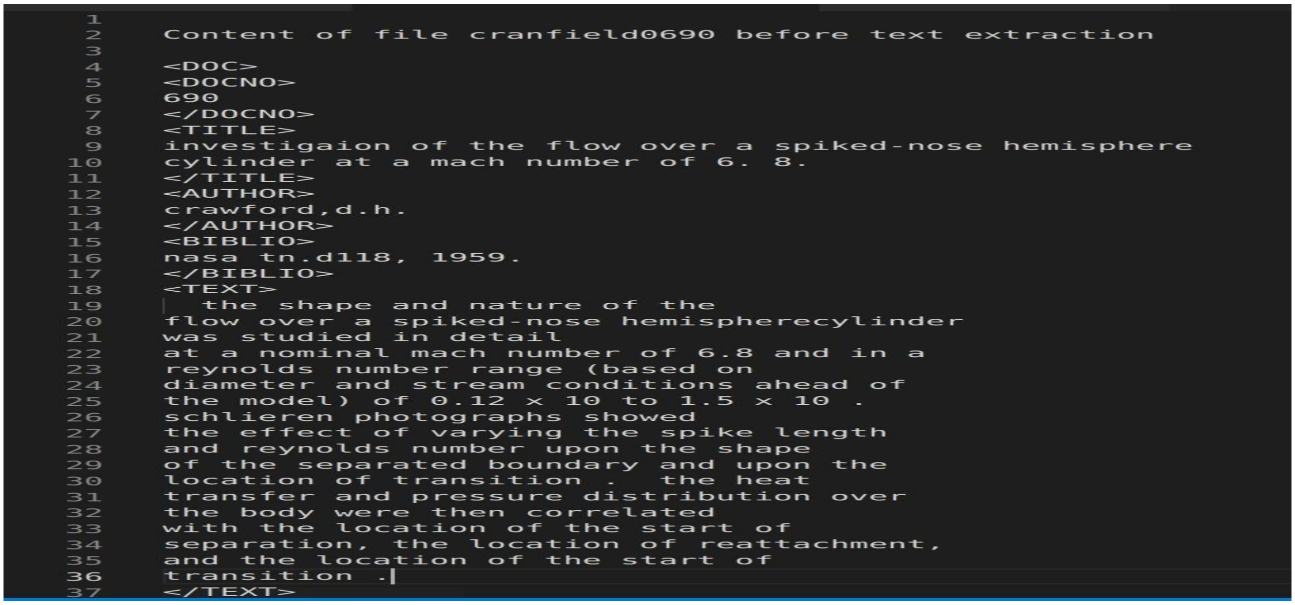
2. Jaccard Coefficient

- We first pre-processed the query.

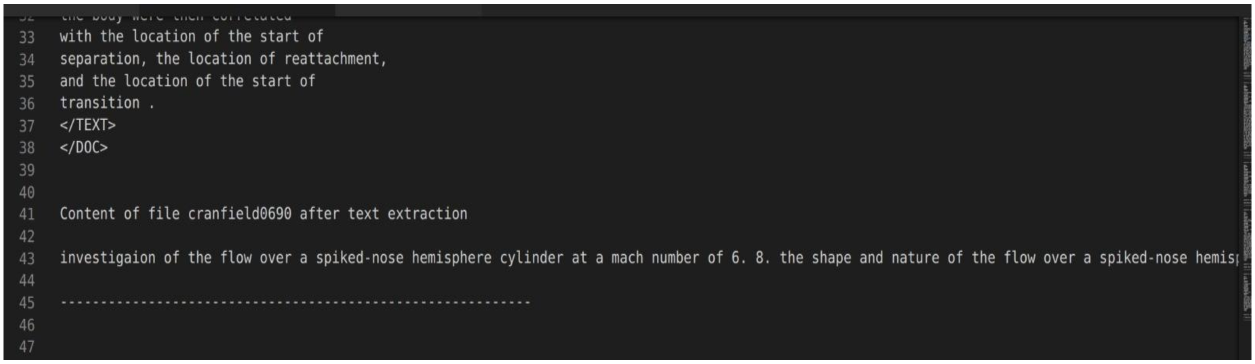
- Then taking each document's pre-processed term list, we found the union and intersection of each document and formed a data frame with each doc id corresponding to its Jaccard coefficient score.

- Then we printed the top 10 documents with their corresponding score.

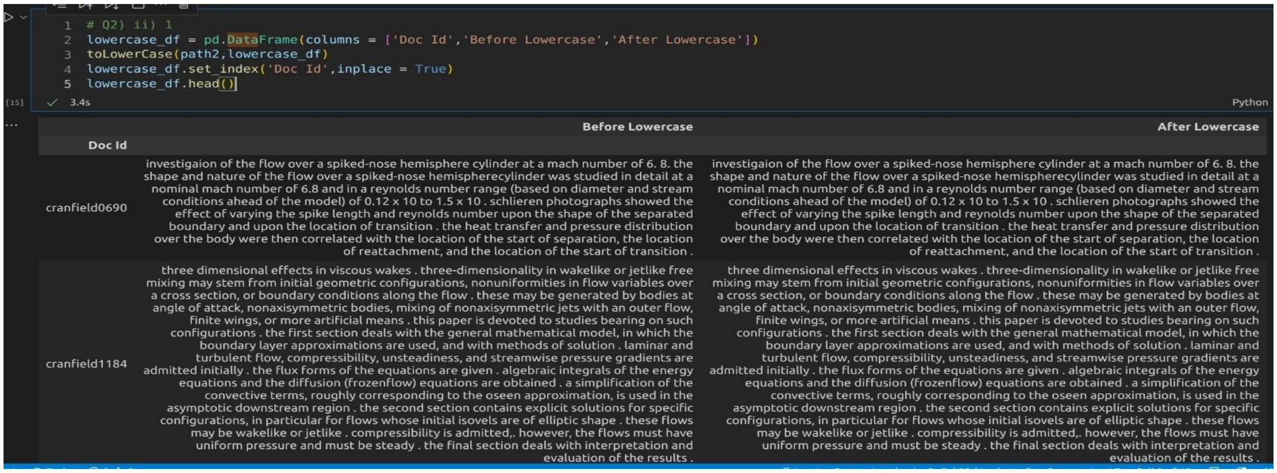
Results:



```

Content of file cranfield0690 before text extraction

<DOC>
<DOCNO>
690
</DOCNO>
<TITLE>
investigaion of the flow over a spiked-nose hemisphere
cylinder at a mach number of 6. 8.
</TITLE>
<AUTHOR>
crawford,d.h.
</AUTHOR>
<BIBLIO>
nasa tn.d118, 1959.
</BIBLIO>
<TEXT>
  the shape and nature of the
flow over a spiked-nose hemispherecylinder
was studied in detail
at a nominal mach number of 6.8 and in a
reynolds number range (based on
diameter and stream conditions ahead of
the model) of 0.12 x 10 to 1.5 x 10 .
schlieren photographs showed
the effect of varying the spike length
and reynolds number upon the shape
of the separated boundary and upon the
location of transition .  the heat
transfer and pressure distribution over
the body were then correlated
with the location of the start of
separation, the location of reattachment,
and the location of the start of
transition .
</TEXT>
```

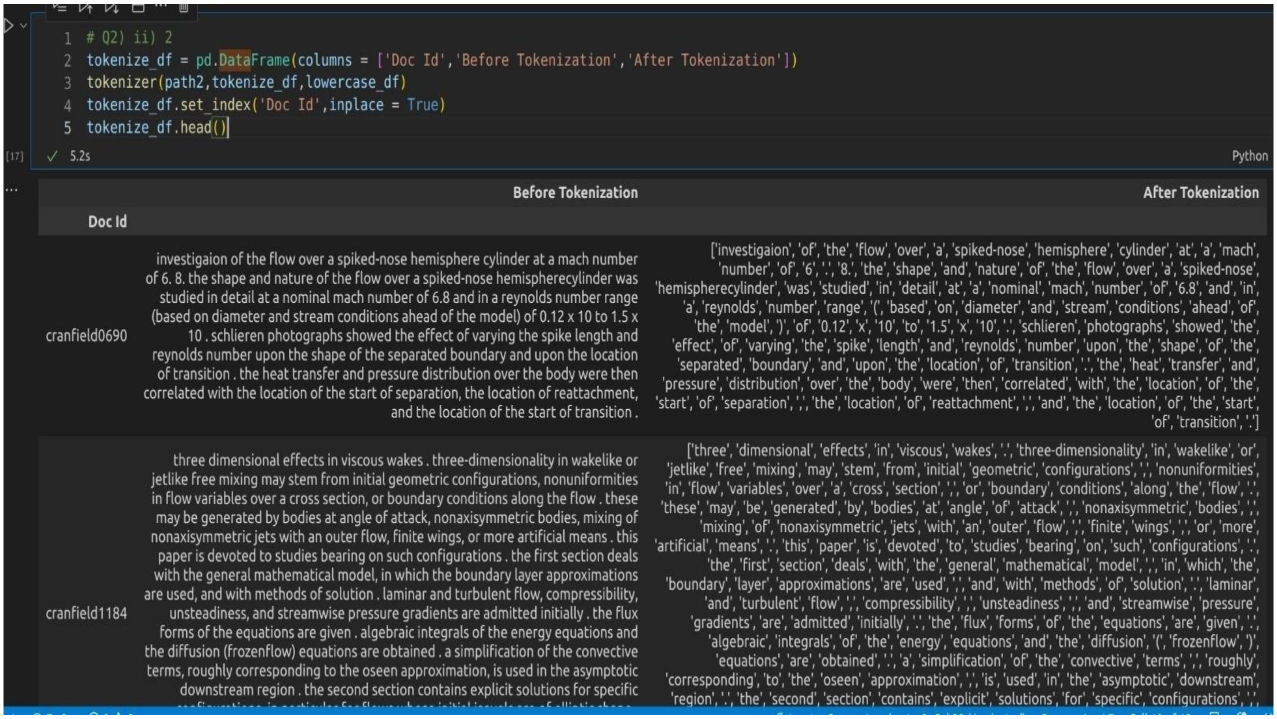


```
the body were then correlated
with the location of the start of
separation, the location of reattachment,
and the location of the start of
transition .
</TEXT>
</DOC>


Content of file cranfield0690 after text extraction

investigaion of the flow over a spiked-nose hemisphere cylinder at a mach number of 6. 8. the shape and nature of the flow over a spiked-nose hemisp

----------------------------------------------------------
```



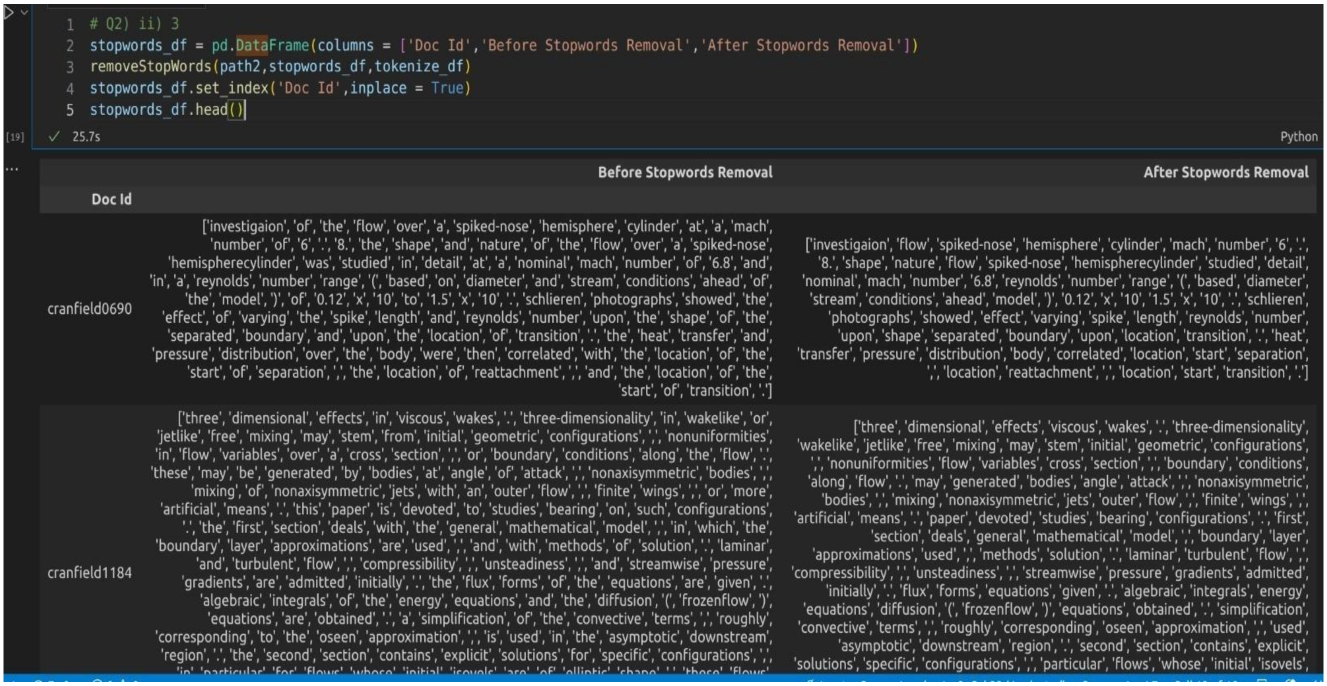
```python
# Q2) ii) 1
lowercase_df = pd.DataFrame(columns = ['Doc Id','Before Lowercase','After Lowercase'])
toLowerCase(path2,lowercase_df)
lowercase_df.set_index('Doc Id',inplace = True)
lowercase_df.head()
```

	Before Lowercase	After Lowercase
Doc Id		
cranfield0690	investigaion of the flow over a spiked-nose hemisphere cylinder at a mach number of 6. 8. the shape and nature of the flow over a spiked-nose hemispherecylinder was studied in detail at a nominal mach number of 6.8 and in a reynolds number range (based on diameter and stream conditions ahead of the model) of 0.12 x 10 to 1.5 x 10 . schlieren photographs showed the effect of varying the spike length and reynolds number upon the shape of the separated boundary and upon the location of transition . the heat transfer and pressure distribution over the body were then correlated with the location of the start of separation, the location of reattachment, and the location of the start of transition .	investigaion of the flow over a spiked-nose hemisphere cylinder at a mach number of 6. 8. the shape and nature of the flow over a spiked-nose hemispherecylinder was studied in detail at a nominal mach number of 6.8 and in a reynolds number range (based on diameter and stream conditions ahead of the model) of 0.12 x 10 to 1.5 x 10 . schlieren photographs showed the effect of varying the spike length and reynolds number upon the shape of the separated boundary and upon the location of transition . the heat transfer and pressure distribution over the body were then correlated with the location of the start of separation, the location of reattachment, and the location of the start of transition .
cranfield1184	three dimensional effects in viscous wakes . three-dimensionality in wakelike or jetlike free mixing may stem from initial geometric configurations, nonuniformities in flow variables over a cross section, or boundary conditions along the flow . these may be generated by bodies at angle of attack, nonaxisymmetric bodies, mixing of nonaxisymmetric jets with an outer flow, finite wings, or more artificial means . this paper is devoted to studies bearing on such configurations . the first section deals with the general mathematical model, in which the boundary layer approximations are used, and with methods of solution . laminar and turbulent flow, compressibility, unsteadiness, and streamwise pressure gradients are admitted initially . the flux forms of the equations are given . algebraic integrals of the energy equations and the diffusion (frozenflow) equations are obtained . a simplification of the convective terms, roughly corresponding to the oseen approximation, is used in the asymptotic downstream region . the second section contains explicit solutions for specific configurations, in particular for flows whose initial isovels are of elliptic shape . these flows may be wakelike or jetlike . compressibility is admitted,. however, the flows must have uniform pressure and must be steady . the final section deals with interpretation and evaluation of the results .	three dimensional effects in viscous wakes . three-dimensionality in wakelike or jetlike free mixing may stem from initial geometric configurations, nonuniformities in flow variables over a cross section, or boundary conditions along the flow . these may be generated by bodies at angle of attack, nonaxisymmetric bodies, mixing of nonaxisymmetric jets with an outer flow, finite wings, or more artificial means . this paper is devoted to studies bearing on such configurations . the first section deals with the general mathematical model, in which the boundary layer approximations are used, and with methods of solution . laminar and turbulent flow, compressibility, unsteadiness, and streamwise pressure gradients are admitted initially . the flux forms of the equations are given . algebraic integrals of the energy equations and the diffusion (frozenflow) equations are obtained . a simplification of the convective terms, roughly corresponding to the oseen approximation, is used in the asymptotic downstream region . the second section contains explicit solutions for specific configurations, in particular for flows whose initial isovels are of elliptic shape . these flows may be wakelike or jetlike . compressibility is admitted,. however, the flows must have uniform pressure and must be steady . the final section deals with interpretation and evaluation of the results .



```python
# Q2) ii) 2
tokenize_df = pd.DataFrame(columns = ['Doc Id','Before Tokenization','After Tokenization'])
tokenizer(path2,tokenize_df,lowercase_df)
tokenize_df.set_index('Doc Id',inplace = True)
tokenize_df.head()
```

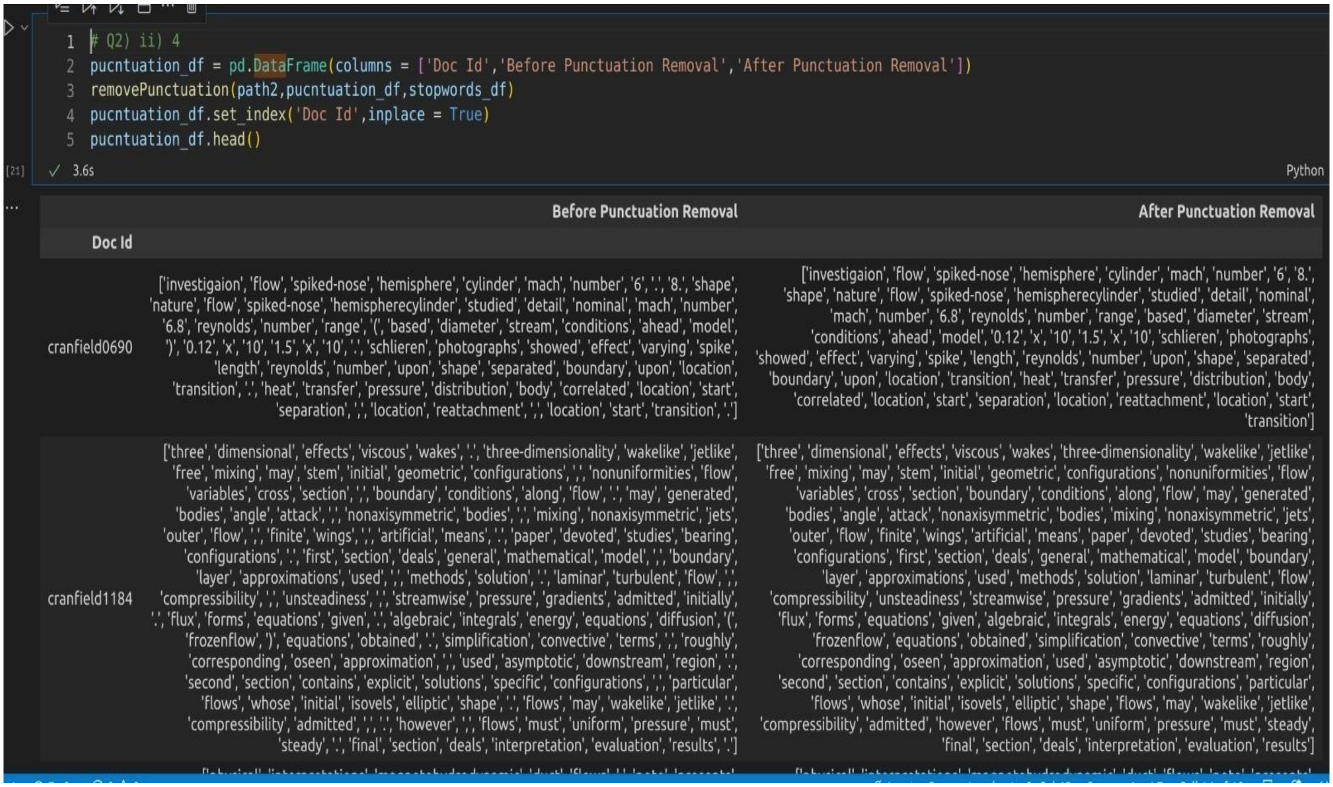
	Before Tokenization	After Tokenization
Doc Id		
cranfield0690	investigaion of the flow over a spiked-nose hemisphere cylinder at a mach number of 6. 8. the shape and nature of the flow over a spiked-nose hemispherecylinder was studied in detail at a nominal mach number of 6.8 and in a reynolds number range (based on diameter and stream conditions ahead of the model) of 0.12 x 10 to 1.5 x 10 . schlieren photographs showed the effect of varying the spike length and reynolds number upon the shape of the separated boundary and upon the location of transition . the heat transfer and pressure distribution over the body were then correlated with the location of the start of separation, the location of reattachment, and the location of the start of transition .	['investigaion', 'of', 'the', 'flow', 'over', 'a', 'spiked-nose', 'hemisphere', 'cylinder', 'at', 'a', 'mach', 'number', 'of', '6', '.', '8.', 'the', 'shape', 'and', 'nature', 'of', 'the', 'flow', 'over', 'a', 'spiked-nose', 'hemispherecylinder', 'was', 'studied', 'in', 'detail', 'at', 'a', 'nominal', 'mach', 'number', 'of', '6.8', 'and', 'in', 'a', 'reynolds', 'number', 'range', '(', 'based', 'on', 'diameter', 'and', 'stream', 'conditions', 'ahead', 'of', 'the', 'model', ')', 'of', '0.12', 'x', '10', 'to', '1.5', 'x', '10', '.', 'schlieren', 'photographs', 'showed', 'the', 'effect', 'of', 'varying', 'the', 'spike', 'length', 'and', 'reynolds', 'number', 'upon', 'the', 'shape', 'of', 'the', 'separated', 'boundary', 'and', 'upon', 'the', 'location', 'of', 'transition', '.', 'the', 'heat', 'transfer', 'and', 'pressure', 'distribution', 'over', 'the', 'body', 'were', 'then', 'correlated', 'with', 'the', 'location', 'of', 'the', 'start', 'of', 'separation', ',', 'the', 'location', 'of', 'reattachment', ',', 'and', 'the', 'location', 'of', 'the', 'start', 'of', 'transition', '.']
cranfield1184	three dimensional effects in viscous wakes . three-dimensionality in wakelike or jetlike free mixing may stem from initial geometric configurations, nonuniformities in flow variables over a cross section, or boundary conditions along the flow . these may be generated by bodies at angle of attack, nonaxisymmetric bodies, mixing of nonaxisymmetric jets with an outer flow, finite wings, or more artificial means . this paper is devoted to studies bearing on such configurations . the first section deals with the general mathematical model, in which the boundary layer approximations are used, and with methods of solution . laminar and turbulent flow, compressibility, unsteadiness, and streamwise pressure gradients are admitted initially . the flux forms of the equations are given . algebraic integrals of the energy equations and the diffusion (frozenflow) equations are obtained . a simplification of the convective terms, roughly corresponding to the oseen approximation, is used in the asymptotic downstream region . the second section contains explicit solutions for specific	['three', 'dimensional', 'effects', 'in', 'viscous', 'wakes', '.', 'three-dimensionality', 'in', 'wakelike', 'or', 'jetlike', 'free', 'mixing', 'may', 'stem', 'from', 'initial', 'geometric', 'configurations', ',', 'nonuniformities', 'in', 'flow', 'variables', 'over', 'a', 'cross', 'section', ',', 'or', 'boundary', 'conditions', 'along', 'the', 'flow', '.', 'these', 'may', 'be', 'generated', 'by', 'bodies', 'at', 'angle', 'of', 'attack', ',', 'nonaxisymmetric', 'bodies', ',', 'mixing', 'of', 'nonaxisymmetric', 'jets', 'with', 'an', 'outer', 'flow', ',', 'finite', 'wings', ',', 'or', 'more', 'artificial', 'means', '.', 'this', 'paper', 'is', 'devoted', 'to', 'studies', 'bearing', 'on', 'such', 'configurations', '.', 'the', 'first', 'section', 'deals', 'with', 'the', 'general', 'mathematical', 'model', ',', 'in', 'which', 'the', 'boundary', 'layer', 'approximations', 'are', 'used', ',', 'and', 'with', 'methods', 'of', 'solution', '.', 'laminar', 'and', 'turbulent', 'flow', ',', 'compressibility', ',', 'unsteadiness', ',', 'and', 'streamwise', 'pressure', 'gradients', 'are', 'admitted', 'initially', '.', 'the', 'flux', 'forms', 'of', 'the', 'equations', 'are', 'given', '.', 'algebraic', 'integrals', 'of', 'the', 'energy', 'equations', 'and', 'the', 'diffusion', '(', 'frozenflow', ')', 'equations', 'are', 'obtained', '.', 'a', 'simplification', 'of', 'the', 'convective', 'terms', ',', 'roughly', 'corresponding', 'to', 'the', 'oseen', 'approximation', ',', 'is', 'used', 'in', 'the', 'asymptotic', 'downstream', 'region', '.', 'the', 'second', 'section', 'contains', 'explicit', 'solutions', 'for', 'specific', 'configurations', ',',



```
# Q2) ii) 3
stopwords_df = pd.DataFrame(columns = ['Doc Id','Before Stopwords Removal','After Stopwords Removal'])
removeStopWords(path2,stopwords_df,tokenize_df)
stopwords_df.set_index('Doc Id',inplace = True)
stopwords_df.head()
```

[19] ✓ 25.7s Python

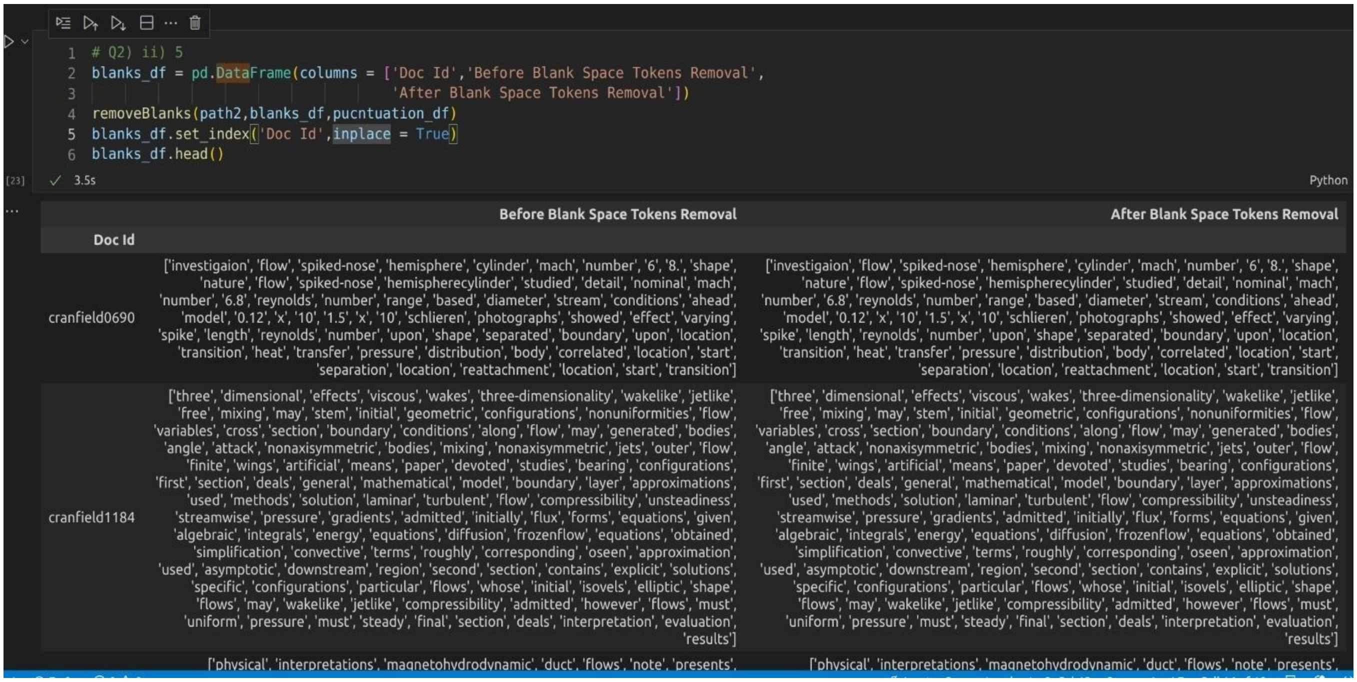
	Before Stopwords Removal	After Stopwords Removal
Doc Id		
cranfield0690	['investigaion', 'of', 'the', 'flow', 'over', 'a', 'spiked-nose', 'hemisphere', 'cylinder', 'at', 'a', 'mach', 'number', 'of', '6', '.', '8.', 'the', 'shape', 'and', 'nature', 'of', 'the', 'flow', 'over', 'a', 'spiked-nose', 'hemispherecylinder', 'was', 'studied', 'in', 'detail', 'at', 'a', 'nominal', 'mach', 'number', 'of', '6.8', 'and', 'in', 'a', 'reynolds', 'number', 'range', '(', 'based', 'on', 'diameter', 'and', 'stream', 'conditions', 'ahead', 'of', 'the', 'model', ')', 'of', '0.12', 'x', '10', 'to', '1.5', 'x', '10', '.', 'schlieren', 'photographs', 'showed', 'the', 'effect', 'of', 'varying', 'the', 'spike', 'length', 'and', 'reynolds', 'number', 'upon', 'the', 'shape', 'of', 'the', 'separated', 'boundary', 'and', 'upon', 'the', 'location', 'of', 'transition', '.', 'the', 'heat', 'transfer', 'and', 'pressure', 'distribution', 'over', 'the', 'body', 'were', 'then', 'correlated', 'with', 'the', 'location', 'of', 'the', 'start', 'of', 'separation', ',', 'the', 'location', 'of', 'reattachment', ',', 'and', 'the', 'location', 'of', 'the', 'start', 'of', 'transition', '.']	['investigaion', 'flow', 'spiked-nose', 'hemisphere', 'cylinder', 'mach', 'number', '6', '.', '8.', 'shape', 'nature', 'flow', 'spiked-nose', 'hemispherecylinder', 'studied', 'detail', 'nominal', 'mach', 'number', '6.8', 'reynolds', 'number', 'range', '(', 'based', 'diameter', 'stream', 'conditions', 'ahead', 'model', ')', '0.12', 'x', '10', '1.5', 'x', '10', '.', 'schlieren', 'photographs', 'showed', 'effect', 'varying', 'spike', 'length', 'reynolds', 'number', 'upon', 'shape', 'separated', 'boundary', 'upon', 'location', 'transition', '.', 'heat', 'transfer', 'pressure', 'distribution', 'body', 'correlated', 'location', 'start', 'separation', ',', 'location', 'reattachment', ',', 'location', 'start', 'transition', '.']
cranfield1184	['three', 'dimensional', 'effects', 'in', 'viscous', 'wakes', '.', 'three-dimensionality', 'in', 'wakelike', 'or', 'jetlike', 'free', 'mixing', 'may', 'stem', 'from', 'initial', 'geometric', 'configurations', ',', 'nonuniformities', 'in', 'flow', 'variables', 'over', 'a', 'cross', 'section', ',', 'or', 'boundary', 'conditions', 'along', 'the', 'flow', '.', 'these', 'may', 'be', 'generated', 'by', 'bodies', 'at', 'angle', 'of', 'attack', ',', 'nonaxisymmetric', 'bodies', ',', 'mixing', 'of', 'nonaxisymmetric', 'jets', 'with', 'an', 'outer', 'flow', ',', 'finite', 'wings', ',', 'or', 'more', 'artificial', 'means', '.', 'this', 'paper', 'is', 'devoted', 'to', 'studies', 'bearing', 'on', 'such', 'configurations', '.', 'the', 'first', 'section', 'deals', 'with', 'the', 'general', 'mathematical', 'model', ',', 'in', 'which', 'the', 'boundary', 'layer', 'approximations', 'are', 'used', ',', 'and', 'with', 'methods', 'of', 'solution', '.', 'laminar', 'and', 'turbulent', 'flow', ',', 'compressibility', ',', 'unsteadiness', ',', 'and', 'streamwise', 'pressure', 'gradients', 'are', 'admitted', 'initially', '.', 'the', 'flux', 'forms', 'of', 'the', 'equations', 'are', 'given', '.', 'algebraic', 'integrals', 'of', 'the', 'energy', 'equations', 'and', 'the', 'diffusion', '(', 'frozenflow', ')', 'equations', 'are', 'obtained', '.', 'a', 'simplification', 'of', 'the', 'convective', 'terms', ',', 'roughly', 'corresponding', 'to', 'the', 'oseen', 'approximation', ',', 'is', 'used', 'in', 'the', 'asymptotic', 'downstream', 'region', '.', 'the', 'second', 'section', 'contains', 'explicit', 'solutions', 'for', 'specific', 'configurations', ',', 'in', 'particular', 'for', 'flows', 'whose', 'initial', 'isovels', 'are', 'of', 'elliptic', 'shape', '.', 'these', 'flows',	['three', 'dimensional', 'effects', 'viscous', 'wakes', '.', 'three-dimensionality', 'wakelike', 'jetlike', 'free', 'mixing', 'may', 'stem', 'initial', 'geometric', 'configurations', ',', 'nonuniformities', 'flow', 'variables', 'cross', 'section', ',', 'boundary', 'conditions', 'along', 'flow', '.', 'may', 'generated', 'bodies', 'angle', 'attack', ',', 'nonaxisymmetric', 'bodies', ',', 'mixing', 'nonaxisymmetric', 'jets', 'outer', 'flow', ',', 'finite', 'wings', ',', 'artificial', 'means', '.', 'paper', 'devoted', 'studies', 'bearing', 'configurations', '.', 'first', 'section', 'deals', 'general', 'mathematical', 'model', ',', 'boundary', 'layer', 'approximations', 'used', ',', 'methods', 'solution', '.', 'laminar', 'turbulent', 'flow', ',', 'compressibility', ',', 'unsteadiness', ',', 'streamwise', 'pressure', 'gradients', 'admitted', 'initially', '.', 'flux', 'forms', 'equations', 'given', '.', 'algebraic', 'integrals', 'energy', 'equations', 'diffusion', '(', 'frozenflow', ')', 'equations', 'obtained', '.', 'simplification', 'convective', 'terms', ',', 'roughly', 'corresponding', 'oseen', 'approximation', ',', 'used', 'asymptotic', 'downstream', 'region', '.', 'second', 'section', 'contains', 'explicit', 'solutions', 'specific', 'configurations', ',', 'particular', 'flows', 'whose', 'initial', 'isovels',



```
# Q2) ii) 4
pucntuation_df = pd.DataFrame(columns = ['Doc Id','Before Punctuation Removal','After Punctuation Removal'])
removePunctuation(path2,pucntuation_df,stopwords_df)
pucntuation_df.set_index('Doc Id',inplace = True)
pucntuation_df.head()
```

[21] ✓ 3.6s Python

	Before Punctuation Removal	After Punctuation Removal
Doc Id		
cranfield0690	['investigaion', 'flow', 'spiked-nose', 'hemisphere', 'cylinder', 'mach', 'number', '6', '.', '8.', 'shape', 'nature', 'flow', 'spiked-nose', 'hemispherecylinder', 'studied', 'detail', 'nominal', 'mach', 'number', '6.8', 'reynolds', 'number', 'range', '(', 'based', 'diameter', 'stream', 'conditions', 'ahead', 'model', ')', '0.12', 'x', '10', '1.5', 'x', '10', '.', 'schlieren', 'photographs', 'showed', 'effect', 'varying', 'spike', 'length', 'reynolds', 'number', 'upon', 'shape', 'separated', 'boundary', 'upon', 'location', 'transition', '.', 'heat', 'transfer', 'pressure', 'distribution', 'body', 'correlated', 'location', 'start', 'separation', ',', 'location', 'reattachment', ',', 'location', 'start', 'transition', '.']	['investigaion', 'flow', 'spiked-nose', 'hemisphere', 'cylinder', 'mach', 'number', '6', '8.', 'shape', 'nature', 'flow', 'spiked-nose', 'hemispherecylinder', 'studied', 'detail', 'nominal', 'mach', 'number', '6.8', 'reynolds', 'number', 'range', 'based', 'diameter', 'stream', 'conditions', 'ahead', 'model', '0.12', 'x', '10', '1.5', 'x', '10', 'schlieren', 'photographs', 'showed', 'effect', 'varying', 'spike', 'length', 'reynolds', 'number', 'upon', 'shape', 'separated', 'boundary', 'upon', 'location', 'transition', 'heat', 'transfer', 'pressure', 'distribution', 'body', 'correlated', 'location', 'start', 'separation', 'location', 'reattachment', 'location', 'start', 'transition']
cranfield1184	['three', 'dimensional', 'effects', 'viscous', 'wakes', '.', 'three-dimensionality', 'wakelike', 'jetlike', 'free', 'mixing', 'may', 'stem', 'initial', 'geometric', 'configurations', ',', 'nonuniformities', 'flow', 'variables', 'cross', 'section', ',', 'boundary', 'conditions', 'along', 'flow', '.', 'may', 'generated', 'bodies', 'angle', 'attack', ',', 'nonaxisymmetric', 'bodies', ',', 'mixing', 'nonaxisymmetric', 'jets', 'outer', 'flow', ',', 'finite', 'wings', ',', 'artificial', 'means', '.', 'paper', 'devoted', 'studies', 'bearing', 'configurations', '.', 'first', 'section', 'deals', 'general', 'mathematical', 'model', ',', 'boundary', 'layer', 'approximations', 'used', ',', 'methods', 'solution', '.', 'laminar', 'turbulent', 'flow', ',', 'compressibility', ',', 'unsteadiness', ',', 'streamwise', 'pressure', 'gradients', 'admitted', 'initially', '.', 'flux', 'forms', 'equations', 'given', '.', 'algebraic', 'integrals', 'energy', 'equations', 'diffusion', '(', 'frozenflow', ')', 'equations', 'obtained', '.', 'simplification', 'convective', 'terms', ',', 'roughly', 'corresponding', 'oseen', 'approximation', ',', 'used', 'asymptotic', 'downstream', 'region', '.', 'second', 'section', 'contains', 'explicit', 'solutions', 'specific', 'configurations', ',', 'particular', 'flows', 'whose', 'initial', 'isovels', 'elliptic', 'shape', '.', 'flows', 'may', 'wakelike', 'jetlike', '.', 'compressibility', 'admitted', ',', '.', 'however', ',', 'flows', 'must', 'uniform', 'pressure', 'must', 'steady', '.', 'final', 'section', 'deals', 'interpretation', 'evaluation', 'results', '.']	['three', 'dimensional', 'effects', 'viscous', 'wakes', 'three-dimensionality', 'wakelike', 'jetlike', 'free', 'mixing', 'may', 'stem', 'initial', 'geometric', 'configurations', 'nonuniformities', 'flow', 'variables', 'cross', 'section', 'boundary', 'conditions', 'along', 'flow', 'may', 'generated', 'bodies', 'angle', 'attack', 'nonaxisymmetric', 'bodies', 'mixing', 'nonaxisymmetric', 'jets', 'outer', 'flow', 'finite', 'wings', 'artificial', 'means', 'paper', 'devoted', 'studies', 'bearing', 'configurations', 'first', 'section', 'deals', 'general', 'mathematical', 'model', 'boundary', 'layer', 'approximations', 'used', 'methods', 'solution', 'laminar', 'turbulent', 'flow', 'compressibility', 'unsteadiness', 'streamwise', 'pressure', 'gradients', 'admitted', 'initially', 'flux', 'forms', 'equations', 'given', 'algebraic', 'integrals', 'energy', 'equations', 'diffusion', 'frozenflow', 'equations', 'obtained', 'simplification', 'convective', 'terms', 'roughly', 'corresponding', 'oseen', 'approximation', 'used', 'asymptotic', 'downstream', 'region', 'second', 'section', 'contains', 'explicit', 'solutions', 'specific', 'configurations', 'particular', 'flows', 'whose', 'initial', 'isovels', 'elliptic', 'shape', 'flows', 'may', 'wakelike', 'jetlike', 'compressibility', 'admitted', 'however', 'flows', 'must', 'uniform', 'pressure', 'must', 'steady', 'final', 'section', 'deals', 'interpretation', 'evaluation', 'results']
	['physical', 'interpretations', 'magnetohydrodynamic', 'duct', 'flows', '.', 'note', 'concepts',	['physical', 'interpretations', 'magnetohydrodynamic', 'duct', 'flows', 'note', 'concepts',

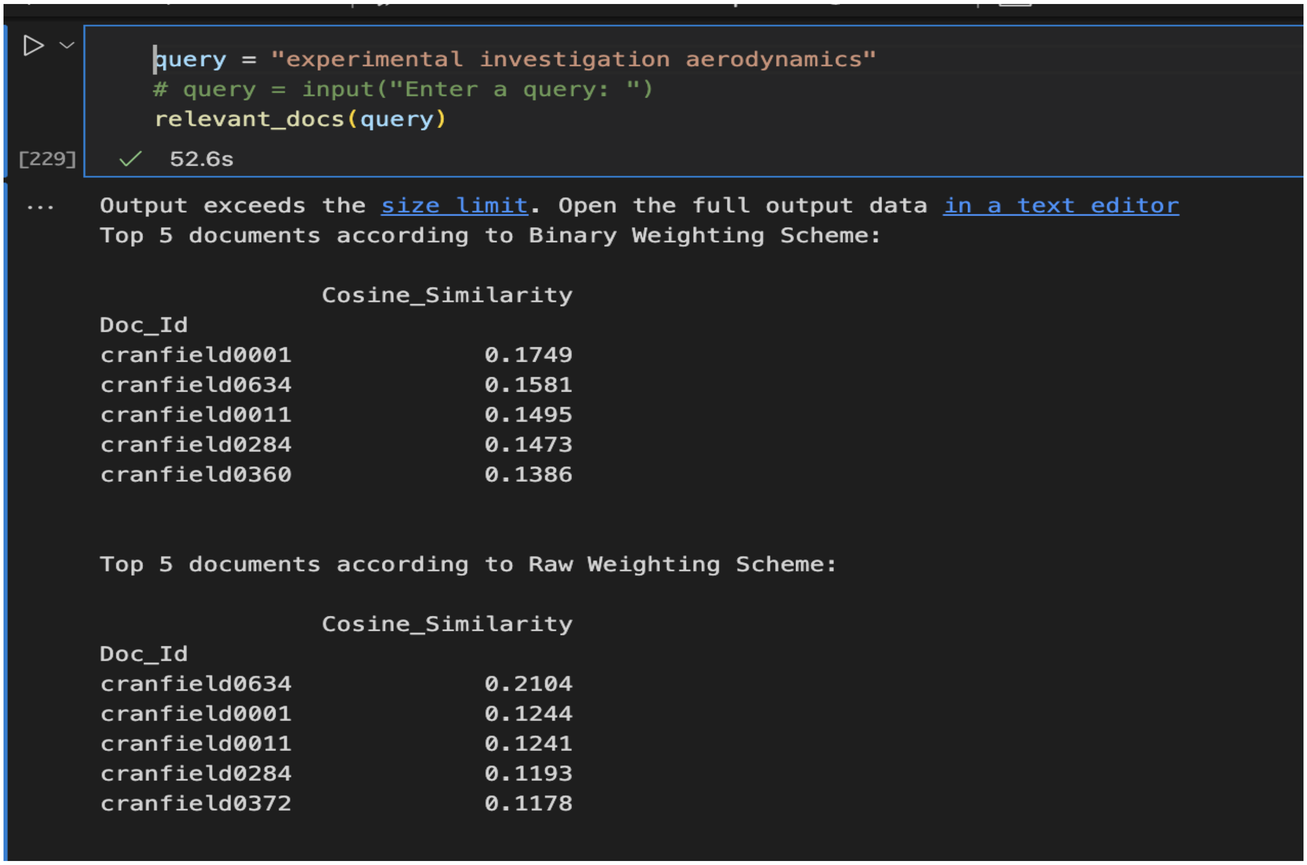


```python
# Q2) ii) 5
blanks_df = pd.DataFrame(columns = ['Doc Id','Before Blank Space Tokens Removal',
                                    'After Blank Space Tokens Removal'])
removeBlanks(path2,blanks_df,pucntuation_df)
blanks_df.set_index('Doc Id',inplace = True)
blanks_df.head()
```

[23] ✓ 3.5s

Python

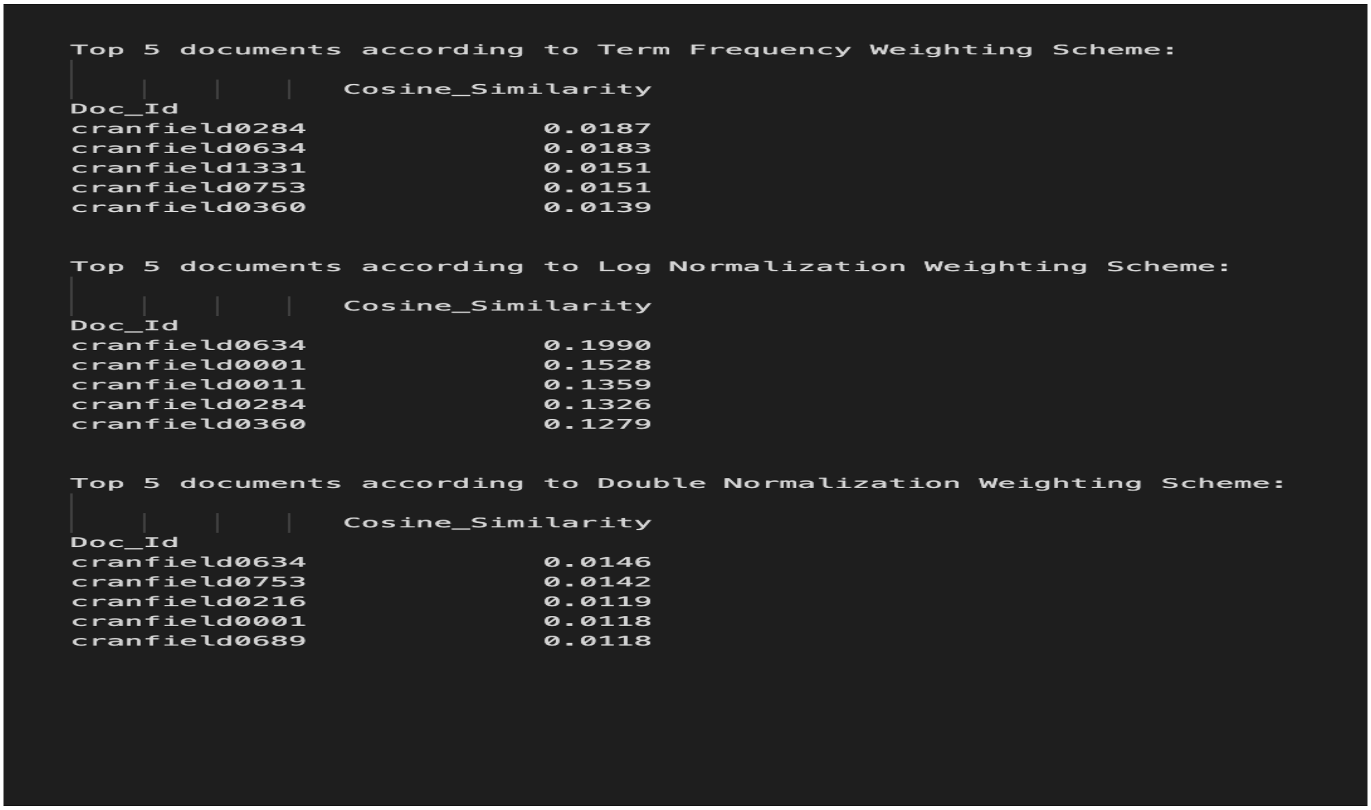
	Before Blank Space Tokens Removal	After Blank Space Tokens Removal
Doc Id		
cranfield0690	['investigaion', 'flow', 'spiked-nose', 'hemisphere', 'cylinder', 'mach', 'number', '6', '8.', 'shape', 'nature', 'flow', 'spiked-nose', 'hemispherecylinder', 'studied', 'detail', 'nominal', 'mach', 'number', '6.8', 'reynolds', 'number', 'range', 'based', 'diameter', 'stream', 'conditions', 'ahead', 'model', '0.12', 'x', '10', '1.5', 'x', '10', 'schlieren', 'photographs', 'showed', 'effect', 'varying', 'spike', 'length', 'reynolds', 'number', 'upon', 'shape', 'separated', 'boundary', 'upon', 'location', 'transition', 'heat', 'transfer', 'pressure', 'distribution', 'body', 'correlated', 'location', 'start', 'separation', 'location', 'reattachment', 'location', 'start', 'transition']	['investigaion', 'flow', 'spiked-nose', 'hemisphere', 'cylinder', 'mach', 'number', '6', '8.', 'shape', 'nature', 'flow', 'spiked-nose', 'hemispherecylinder', 'studied', 'detail', 'nominal', 'mach', 'number', '6.8', 'reynolds', 'number', 'range', 'based', 'diameter', 'stream', 'conditions', 'ahead', 'model', '0.12', 'x', '10', '1.5', 'x', '10', 'schlieren', 'photographs', 'showed', 'effect', 'varying', 'spike', 'length', 'reynolds', 'number', 'upon', 'shape', 'separated', 'boundary', 'upon', 'location', 'transition', 'heat', 'transfer', 'pressure', 'distribution', 'body', 'correlated', 'location', 'start', 'separation', 'location', 'reattachment', 'location', 'start', 'transition']
cranfield1184	['three', 'dimensional', 'effects', 'viscous', 'wakes', 'three-dimensionality', 'wakelike', 'jetlike', 'free', 'mixing', 'may', 'stem', 'initial', 'geometric', 'configurations', 'nonuniformities', 'flow', 'variables', 'cross', 'section', 'boundary', 'conditions', 'along', 'flow', 'may', 'generated', 'bodies', 'angle', 'attack', 'nonaxisymmetric', 'bodies', 'mixing', 'nonaxisymmetric', 'jets', 'outer', 'flow', 'finite', 'wings', 'artificial', 'means', 'paper', 'devoted', 'studies', 'bearing', 'configurations', 'first', 'section', 'deals', 'general', 'mathematical', 'model', 'boundary', 'layer', 'approximations', 'used', 'methods', 'solution', 'laminar', 'turbulent', 'flow', 'compressibility', 'unsteadiness', 'streamwise', 'pressure', 'gradients', 'admitted', 'initially', 'flux', 'forms', 'equations', 'given', 'algebraic', 'integrals', 'energy', 'equations', 'diffusion', 'frozenflow', 'equations', 'obtained', 'simplification', 'convective', 'terms', 'roughly', 'corresponding', 'oseen', 'approximation', 'used', 'asymptotic', 'downstream', 'region', 'second', 'section', 'contains', 'explicit', 'solutions', 'specific', 'configurations', 'particular', 'flows', 'whose', 'initial', 'isovels', 'elliptic', 'shape', 'flows', 'may', 'wakelike', 'jetlike', 'compressibility', 'admitted', 'however', 'flows', 'must', 'uniform', 'pressure', 'must', 'steady', 'final', 'section', 'deals', 'interpretation', 'evaluation', 'results']	['three', 'dimensional', 'effects', 'viscous', 'wakes', 'three-dimensionality', 'wakelike', 'jetlike', 'free', 'mixing', 'may', 'stem', 'initial', 'geometric', 'configurations', 'nonuniformities', 'flow', 'variables', 'cross', 'section', 'boundary', 'conditions', 'along', 'flow', 'may', 'generated', 'bodies', 'angle', 'attack', 'nonaxisymmetric', 'bodies', 'mixing', 'nonaxisymmetric', 'jets', 'outer', 'flow', 'finite', 'wings', 'artificial', 'means', 'paper', 'devoted', 'studies', 'bearing', 'configurations', 'first', 'section', 'deals', 'general', 'mathematical', 'model', 'boundary', 'layer', 'approximations', 'used', 'methods', 'solution', 'laminar', 'turbulent', 'flow', 'compressibility', 'unsteadiness', 'streamwise', 'pressure', 'gradients', 'admitted', 'initially', 'flux', 'forms', 'equations', 'given', 'algebraic', 'integrals', 'energy', 'equations', 'diffusion', 'frozenflow', 'equations', 'obtained', 'simplification', 'convective', 'terms', 'roughly', 'corresponding', 'oseen', 'approximation', 'used', 'asymptotic', 'downstream', 'region', 'second', 'section', 'contains', 'explicit', 'solutions', 'specific', 'configurations', 'particular', 'flows', 'whose', 'initial', 'isovels', 'elliptic', 'shape', 'flows', 'may', 'wakelike', 'jetlike', 'compressibility', 'admitted', 'however', 'flows', 'must', 'uniform', 'pressure', 'must', 'steady', 'final', 'section', 'deals', 'interpretation', 'evaluation', 'results']
	['physical', 'interpretations', 'magnetohydrodynamic', 'duct', 'flows', 'note', 'presents',	['physical', 'interpretations', 'magnetohydrodynamic', 'duct', 'flows', 'note', 'presents',



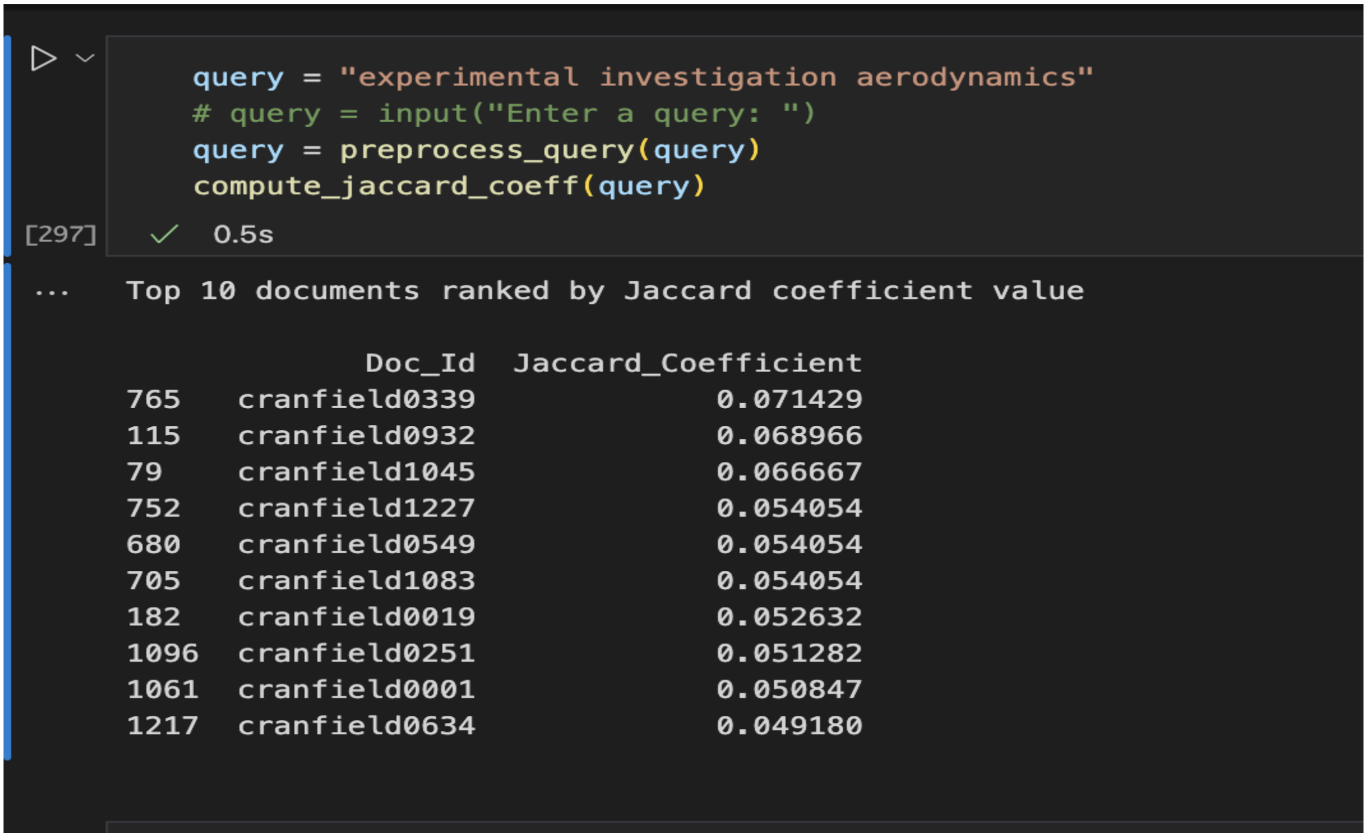
```python
query = "experimental investigation aerodynamics"
# query = input("Enter a query: ")
relevant_docs(query)
```

[229] ✓ 52.6s

```
Output exceeds the size limit. Open the full output data in a text editor
Top 5 documents according to Binary Weighting Scheme:

               Cosine_Similarity
Doc_Id
cranfield0001             0.1749
cranfield0634             0.1581
cranfield0011             0.1495
cranfield0284             0.1473
cranfield0360             0.1386


Top 5 documents according to Raw Weighting Scheme:

               Cosine_Similarity
Doc_Id
cranfield0634             0.2104
cranfield0001             0.1244
cranfield0011             0.1241
cranfield0284             0.1193
cranfield0372             0.1178
```



```
Top 5 documents according to Term Frequency Weighting Scheme:
|   |   |   |   Cosine_Similarity
Doc_Id
cranfield0284              0.0187
cranfield0634              0.0183
cranfield1331              0.0151
cranfield0753              0.0151
cranfield0360              0.0139

Top 5 documents according to Log Normalization Weighting Scheme:
|   |   |   |   Cosine_Similarity
Doc_Id
cranfield0634              0.1990
cranfield0001              0.1528
cranfield0011              0.1359
cranfield0284              0.1326
cranfield0360              0.1279

Top 5 documents according to Double Normalization Weighting Scheme:
|   |   |   |   Cosine_Similarity
Doc_Id
cranfield0634              0.0146
cranfield0753              0.0142
cranfield0216              0.0119
cranfield0001              0.0118
cranfield0689              0.0118
```



```
query = "experimental investigation aerodynamics"
# query = input("Enter a query: ")
query = preprocess_query(query)
compute_jaccard_coeff(query)
```

[297] ✓ 0.5s

```
Top 10 documents ranked by Jaccard coefficient value

             Doc_Id  Jaccard_Coefficient
765   cranfield0339             0.071429
115   cranfield0932             0.068966
79    cranfield1045             0.066667
752   cranfield1227             0.054054
680   cranfield0549             0.054054
705   cranfield1083             0.054054
182   cranfield0019             0.052632
1096  cranfield0251             0.051282
1061  cranfield0001             0.050847
1217  cranfield0634             0.049180
```

Binary Weighting Scheme:

Pros-

1. It can be helpful for tasks like spam filtering and sentiment analysis, where terms' presence and absence are more important than their frequency.
2. It is simple, easy to implement and requires less computation.

Cons-

1. There is a loss of information as it doesn't consider the frequency of the term in each document.
2. It over-simplifies the representation of TF and also ignores the document length.

Raw Weighting Scheme:

Pros-

1. Provides the importance of each term in a document. Terms that appear rarely can be considered more important for that document.
2. It's simple to compute.

Cons-

1. It's not a good metric in cases where documents vary primarily by length.
2. It ignores the ordering of the terms, i.e. each word in a document is considered to be independent.

Term Frequency Weighting Scheme:

Pros-

1. It adjusts the frequency of each term by the document, thus normalising it.
2. Gives more importance to terms appearing more frequently.

Cons-

1. For varying document lengths, it might not be a good metric.
2. It may not take into account the arrangement of the terms.

Log Normalization Weighting Scheme:

Pros-

1. Reduces the problem of zero frequency.
2. Reduces the impact of outliers.

Cons-
1. It can give varying results for different bases of log.
2. In cases where we need the actual frequency of the terms, we need to recompute the actual frequency values, which requires more computation.

Double Normalization Weighting Scheme:

Pros-
1. An advantage where documents are of varying lengths is their use; in this case, longer documents repeat the exact words much more frequently.

Cons-
1. It's hard to tune as a change in the stop word list can majorly affect the term weightings and the ranking.
2. Not helpful in documents containing an outlier term with high frequency, not adding any value to the content.

Q2

Implement a Naive Bayes classifier with TF-ICF (term frequency-inverse category frequency)

weighting scheme to classify documents into predefined categories based on their content.

KNOW THE DATASET :

1. Dataset has three columns named ArticleId, Text, and Category.
2. Total No of document in dataset: 1490 rows * 2 cols

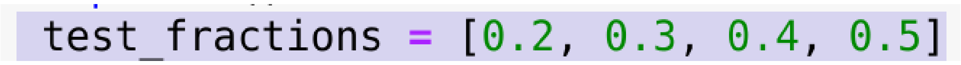
PREPROCESSING:
Following two different approaches were implemented:

Method A:
1. Punctuation removal
2. Lowercasing
3. Stopwords removal
4. Tokenization
5. Lemmatization
6. Integer Removal

Method B:
1. Punctuation removal
2. Lowercasing
3. Stopwords removal
4. Tokenization

5. Stemming
6. Lemmatization
7. Integer Removal

Train Test Splits used:



```
test_fractions = [0.2, 0.3, 0.4, 0.5]
```

Training the Naive Bayes classifier with TF-ICF:

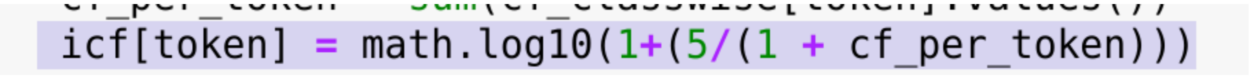
- **Implement the Naive Bayes classifier with the TF-ICF weighting scheme.**

TF-ICF score for a given term belonging to a class can be calculated as follows:
Term Frequency (TF): Number of occurrences of a term in all documents of a particular class
Class Frequency (CF): Number of classes in which that term occurs
Inverse-Class Frequency (ICF): log(N / CF), where N represents the number of classes (log 10)

Some Key - points observations:

```
tf = {key:{term:1 for term in vocab} for key in category}
```

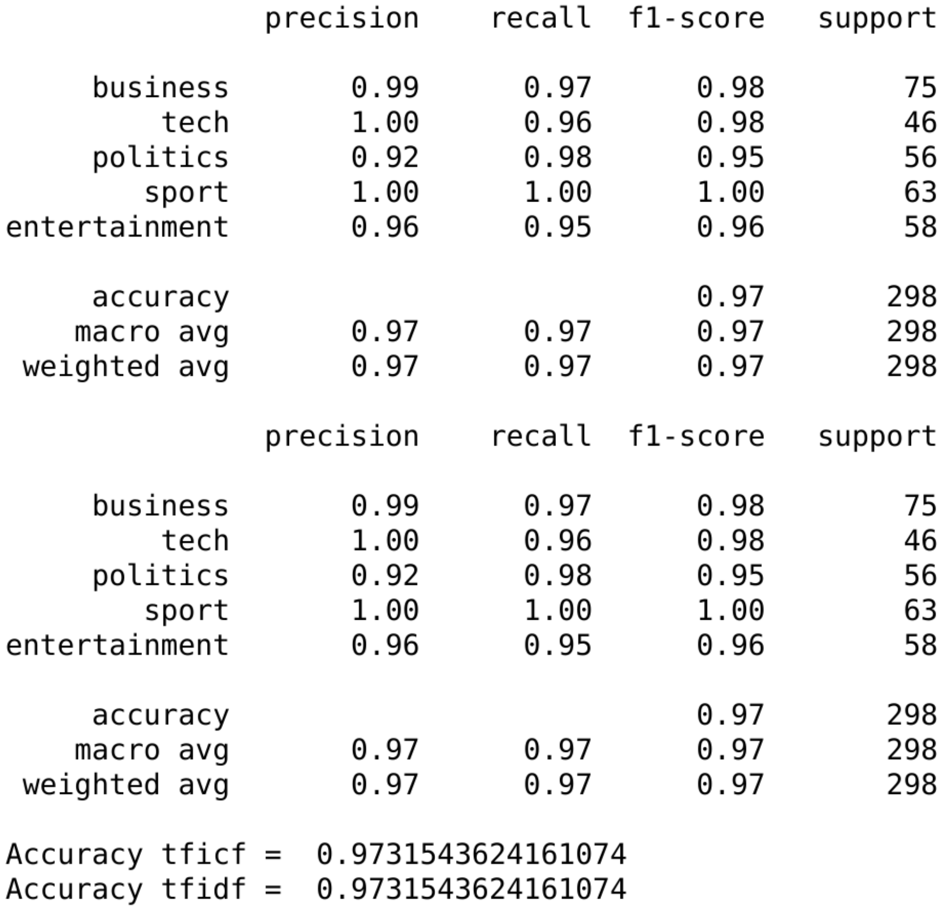
Initialised tf with 1 so as to avoid zeroes later affecting conditional probabilities calculation.



```
icf[token] = math.log10(1+(5/(1 + cf_per_token)))
```

Similarly, for icf we added one on denominator to avoid division by zero; and added 1 overall to maintain positive range values for log.

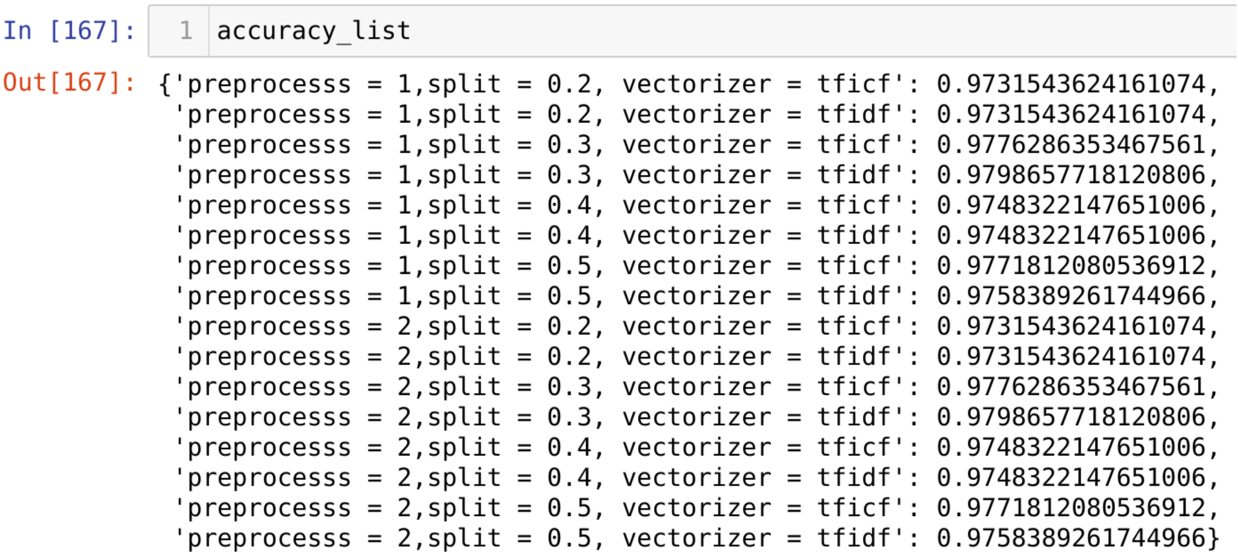
We, are using log likelihood to calculate and add log terms so that product doesn't lose out on values due to underflow.



```
               precision    recall  f1-score   support

     business       0.99      0.97      0.98        75
         tech       1.00      0.96      0.98        46
     politics       0.92      0.98      0.95        56
        sport       1.00      1.00      1.00        63
entertainment       0.96      0.95      0.96        58

     accuracy                           0.97       298
    macro avg       0.97      0.97      0.97       298
 weighted avg       0.97      0.97      0.97       298

               precision    recall  f1-score   support

     business       0.99      0.97      0.98        75
         tech       1.00      0.96      0.98        46
     politics       0.92      0.98      0.95        56
        sport       1.00      1.00      1.00        63
entertainment       0.96      0.95      0.96        58

     accuracy                           0.97       298
    macro avg       0.97      0.97      0.97       298
 weighted avg       0.97      0.97      0.97       298

Accuracy tficf =  0.9731543624161074
Accuracy tfidf =  0.9731543624161074
```

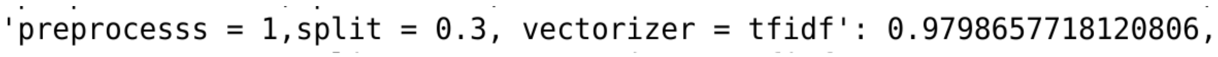
Above is a sample classification report on testing our Naive_Bayes Model on a particular sample of test-split.

Likewise, we conducted 16 experiments with different preprocessing techniques, splits and vectorizer. Below are the following results:



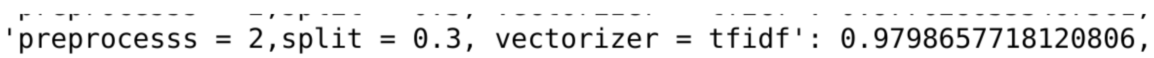
```
In [167]:   1 accuracy_list

Out[167]: {'preprocesss = 1,split = 0.2, vectorizer = tficf': 0.9731543624161074,
            'preprocesss = 1,split = 0.2, vectorizer = tfidf': 0.9731543624161074,
            'preprocesss = 1,split = 0.3, vectorizer = tficf': 0.9776286353467561,
            'preprocesss = 1,split = 0.3, vectorizer = tfidf': 0.9798657718120806,
            'preprocesss = 1,split = 0.4, vectorizer = tficf': 0.9748322147651006,
            'preprocesss = 1,split = 0.4, vectorizer = tfidf': 0.9748322147651006,
            'preprocesss = 1,split = 0.5, vectorizer = tficf': 0.9771812080536912,
            'preprocesss = 1,split = 0.5, vectorizer = tfidf': 0.9758389261744966,
            'preprocesss = 2,split = 0.2, vectorizer = tficf': 0.9731543624161074,
            'preprocesss = 2,split = 0.2, vectorizer = tfidf': 0.9731543624161074,
            'preprocesss = 2,split = 0.3, vectorizer = tficf': 0.9776286353467561,
            'preprocesss = 2,split = 0.3, vectorizer = tfidf': 0.9798657718120806,
            'preprocesss = 2,split = 0.4, vectorizer = tficf': 0.9748322147651006,
            'preprocesss = 2,split = 0.4, vectorizer = tfidf': 0.9748322147651006,
            'preprocesss = 2,split = 0.5, vectorizer = tficf': 0.9771812080536912,
            'preprocesss = 2,split = 0.5, vectorizer = tfidf': 0.9758389261744966}
```

The best reported and improved accuracies are with



```
'preprocesss = 1,split = 0.3, vectorizer = tfidf': 0.9798657718120806,
```

And



```
'preprocesss = 2,split = 0.3, vectorizer = tfidf': 0.9798657718120806,
```

With a 70:30 split, the classifier is giving best accuracies with the tfidf vectorizer.
However, these differences are very little and all the models work reasonably well so all are good classifiers.

Q3

1) ***The first objective is to create a file that rearranges the query-URL pairs in order of the maximum DCG (discounted cumulative gain). The number of such files that could be made should also be stated.***

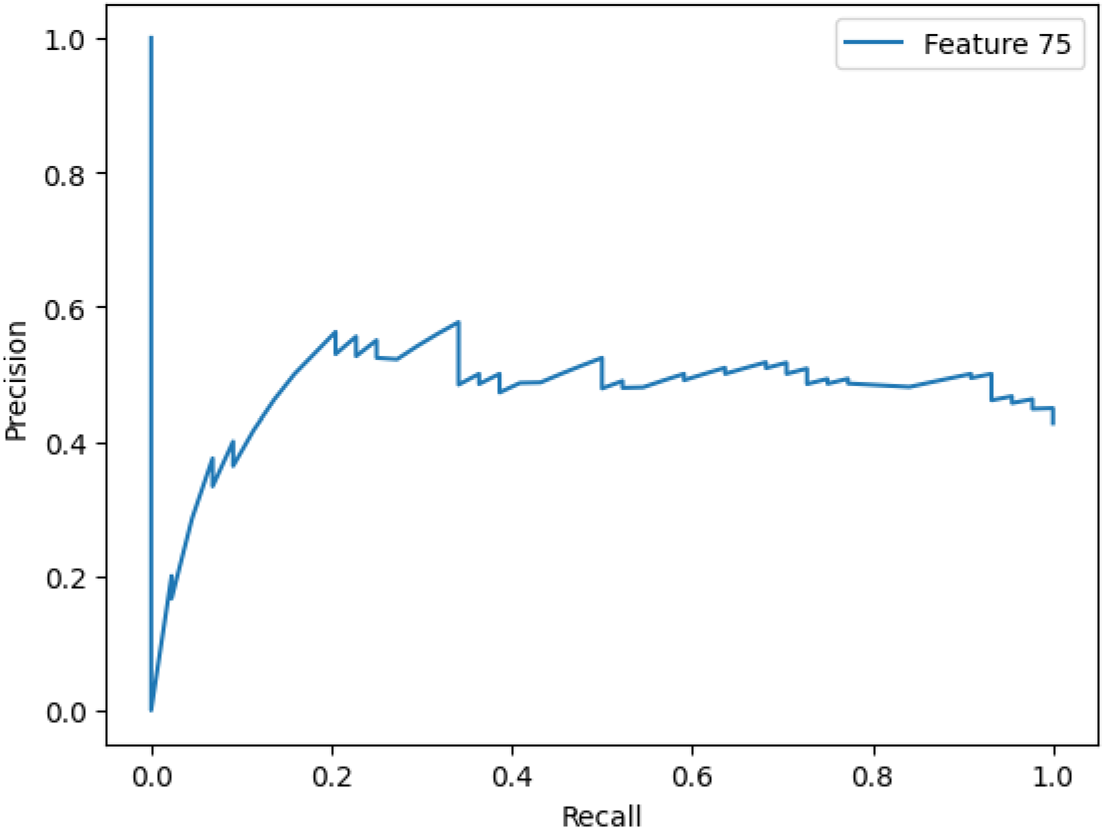
   To create such a file, We need to sort the given document in decreasing order w.r.t the relevance judgement score. The obtained URL order will give us the maximum DCG values. “maxDCG.csv” is the file in my case. No. of such files are 19893497375938370599826047614905329896936840170566570588205180312704857992695193482412686565431050240000000000000000000000. (59! * 26! * 17!)

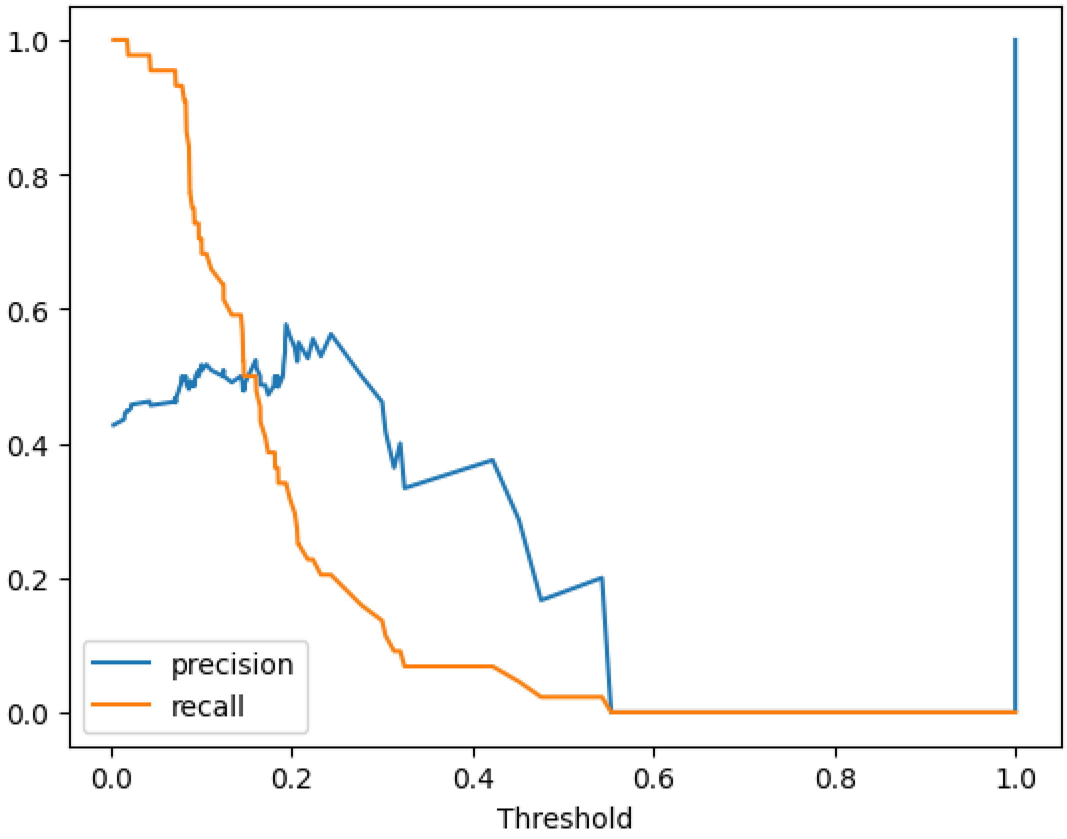
2) ***Next, compute the nDCG (normalized discounted cumulative gain) for the dataset. This involves calculating nDCG at position 50 and for the entire dataset.***

   nDCG = DCG / Max_Possible_DCG

nDCG at position 50 = 0.352
nDCG for entire dataset = 0.597

3) ***Plot a Precision-Recall curve for the query "qid:4". The curve should help visualize the trade-off between precision and recall as the model's threshold for relevance is adjusted.***





From the above curves, we can infer that when the threshold is very low, the recall is very high. This is expected as low threshold means large number of documents will be labeled as relevant, and lot of relevant documents will be retrieved. But when the threshold is very high, recall is very low as very less documents are retrieved and many of the relevant documents are left out.