Homework-5 Report

Rohith Chandra Kandambeth

1. Methodology

a. Obtaining Queries

   i. Description of the assigned queries

   West African Ebola Epidemic: This epidemic refers to the largest outbreak of Ebola virus disease in history, primarily affecting several countries in West Africa from 2014 to 2016, most notably Liberia, Guinea, and Sierra Leone. Characterized by the transmission of the Ebola virus through direct contact with infected bodily fluids, the epidemic led to significant loss of life and economic disruption in the affected regions, prompting international public health responses.

   H1N1 Swine Flu Pandemic: The H1N1 pandemic, which occurred in 2009, was caused by a novel influenza A virus. Originating in Mexico in early 2009 before spreading globally, this pandemic was characterized by the rapid transmission of H1N1, which contained a unique combination of influenza genes not previously identified in animals or people. Despite concerns about its severity, the pandemic's impact was mitigated through widespread vaccination efforts and public health measures, leading to fewer deaths than initially feared.

   COVID-19: COVID-19, caused by the SARS-CoV-2 virus, was first identified in December 2019 in Wuhan, China, and quickly evolved into a global pandemic. Marked by symptoms ranging from mild respiratory issues to severe acute respiratory syndrome, COVID-19's highly contagious nature led to widespread health crises, economic disruptions, and significant mortality worldwide. The pandemic prompted global efforts in vaccination, public health measures, and changes to daily life to control its spread and impact.

   ii. Explanation of the relevance to your vertical search engine topic

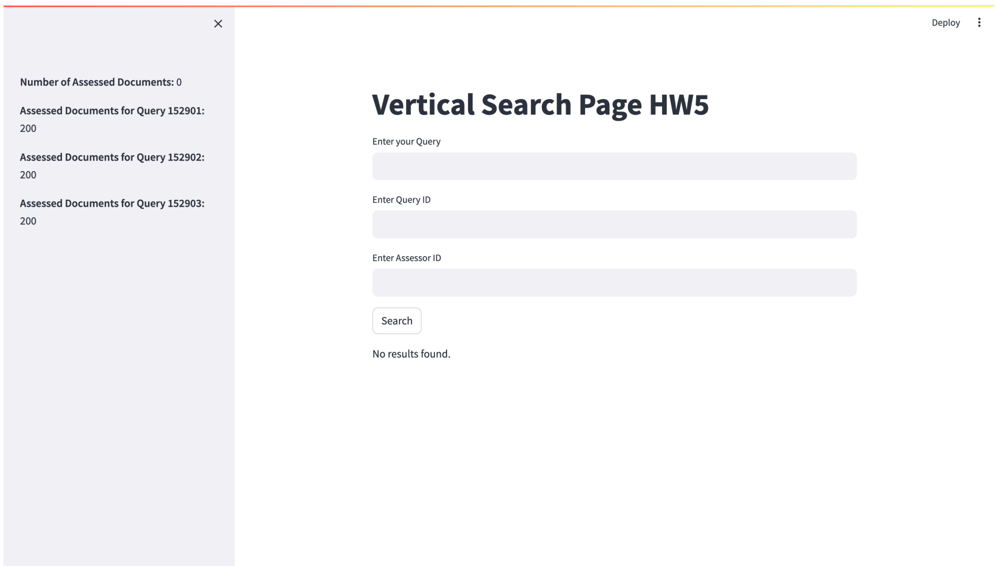
   The relevance is high since the three queries above was used to create the index we are searching using the vertical search engine.

b. Assessment Graphical Interface

   i. Features of the interface

   The interface allows the user to enter the Query, Query Id, and Accessor id. After the search button is clicked 250 results for the query is displayed using es search. Each result can be opened and marked as non-relevant, relevant, or very relevant. On the click of update button, the relevance of each document is saved to a file. The number of documents scored for each query is shown in the side of the interface.
   Additionally, the interface highlights the words in the query.

   ii. Screenshots of the interface


Deploy

Number of Assessed Documents: 0

Assessed Documents for Query 152901: 200

Assessed Documents for Query 152902: 200

Assessed Documents for Query 152903: 200

Vertical Search Page HW5

Enter your Query

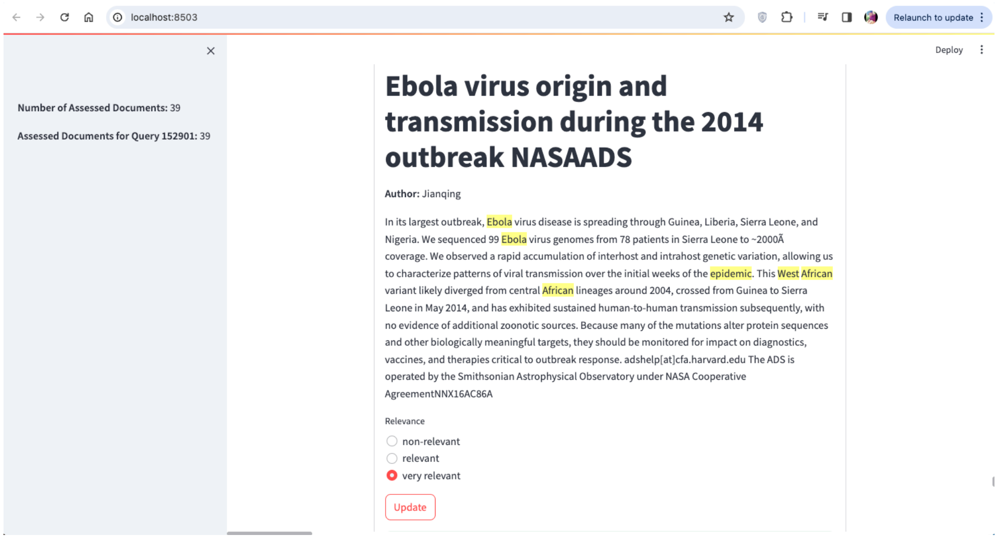
Enter Query ID

Enter Assessor ID

Search

No results found.

Shows 200 for each query because I scored 600 documents already


localhost:8503

Relaunch to update

Deploy

Number of Assessed Documents: 39

Assessed Documents for Query 152901: 39

Ebola virus origin and transmission during the 2014 outbreak NASAADS

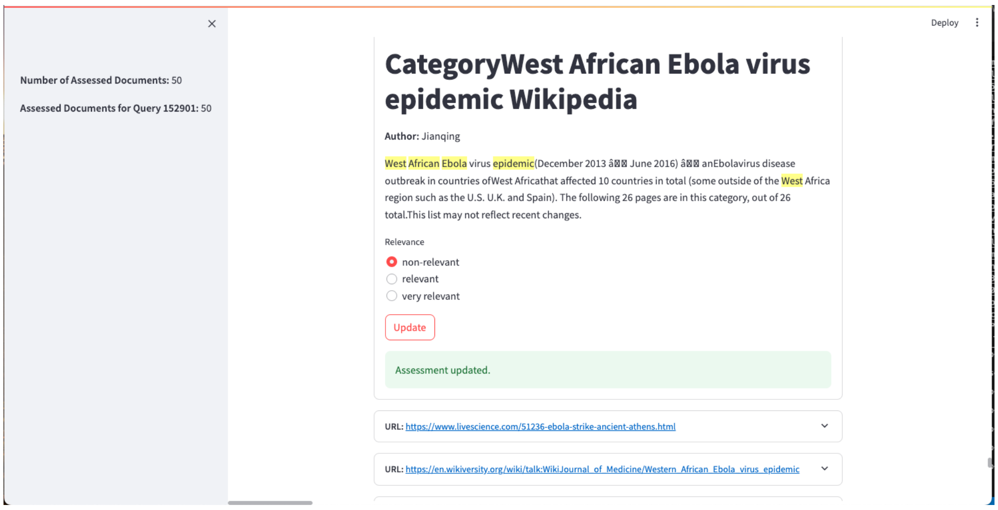
Author: Jianqing

In its largest outbreak, Ebola virus disease is spreading through Guinea, Liberia, Sierra Leone, and Nigeria. We sequenced 99 Ebola virus genomes from 78 patients in Sierra Leone to ~2000Ã coverage. We observed a rapid accumulation of interhost and intrahost genetic variation, allowing us to characterize patterns of viral transmission over the initial weeks of the epidemic. This West African variant likely diverged from central African lineages around 2004, crossed from Guinea to Sierra Leone in May 2014, and has exhibited sustained human-to-human transmission subsequently, with no evidence of additional zoonotic sources. Because many of the mutations alter protein sequences and other biologically meaningful targets, they should be monitored for impact on diagnostics, vaccines, and therapies critical to outbreak response. adshelp[at]cfa.harvard.edu The ADS is operated by the Smithsonian Astrophysical Observatory under NASA Cooperative AgreementNNX16AC86A

Relevance

non-relevant
relevant
[✓ very relevant]

Update


Deploy

Number of Assessed Documents: 50

Assessed Documents for Query 152901: 50

CategoryWest African Ebola virus epidemic Wikipedia

Author: Jianqing

West African Ebola virus epidemic(December 2013 â June 2016) â anEbolavirus disease outbreak in countries ofWest Africathat affected 10 countries in total (some outside of the West Africa region such as the U.S. U.K. and Spain). The following 26 pages are in this category, out of 26 total.This list may not reflect recent changes.

Relevance

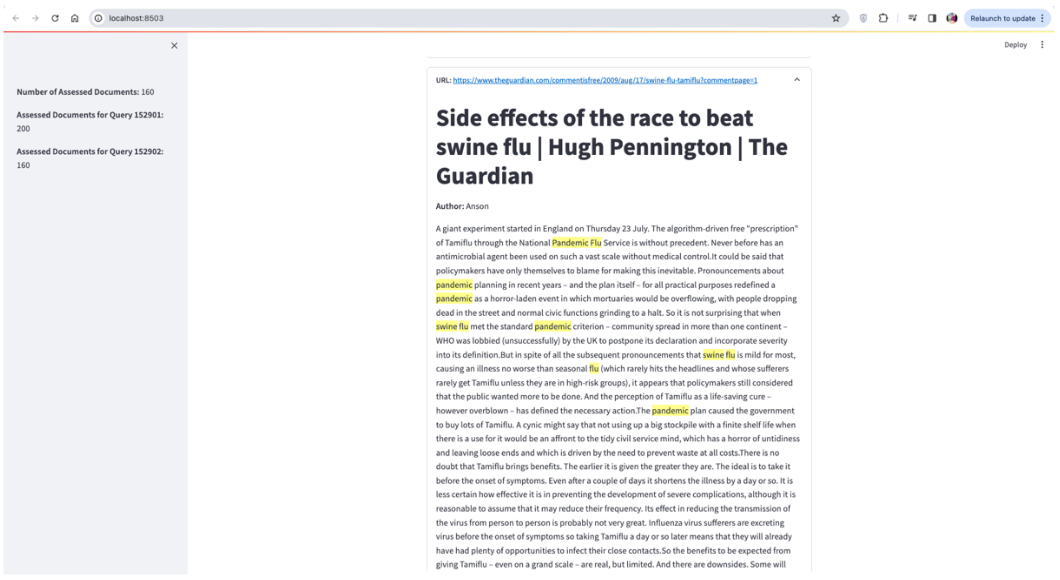
[✓ non-relevant]
relevant
very relevant

Update

Assessment updated.

URL: https://www.livescience.com/51236-ebola-strike-ancient-athens.html

URL: https://en.wikiversity.org/wiki/talk:WikiJournal_of_Medicine/Western_African_Ebola_virus_epidemic



Number of Assessed Documents: 160

Assessed Documents for Query 152901: 200

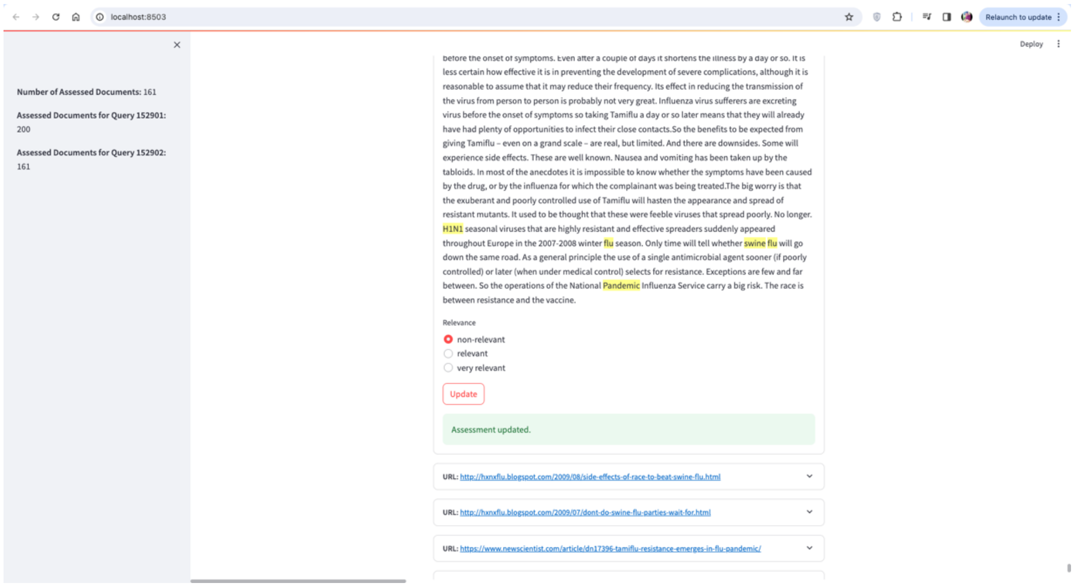
Assessed Documents for Query 152902: 160

URL: https://www.theguardian.com/commentisfree/2009/aug/17/swine-flu-tamiflu?commentpage=1

Side effects of the race to beat swine flu | Hugh Pennington | The Guardian

Author: Anson

A giant experiment started in England on Thursday 23 July. The algorithm-driven free "prescription" of Tamiflu through the National Pandemic Flu Service is without precedent. Never before has an antimicrobial agent been used on such a vast scale without medical control.It could be said that policymakers have only themselves to blame for making this inevitable. Pronouncements about pandemic planning in recent years – and the plan itself – for all practical purposes redefined a pandemic as a horror-laden event in which mortuaries would be overflowing, with people dropping dead in the street and normal civic functions grinding to a halt. So it is not surprising that when swine flu met the standard pandemic criterion – community spread in more than one continent – WHO was lobbied (unsuccessfully) by the UK to postpone its declaration and incorporate severity into its definition.But in spite of all the subsequent pronouncements that swine flu is mild for most, causing an illness no worse than seasonal flu (which rarely hits the headlines and whose sufferers rarely get Tamiflu unless they are in high-risk groups), it appears that policymakers still considered that the public wanted more to be done. And the perception of Tamiflu as a life-saving cure – however overblown – has defined the necessary action.The pandemic plan caused the government to buy lots of Tamiflu. A cynic might say that not using up a big stockpile with a finite shelf life when there is a use for it would be an affront to the tidy civil service mind, which has a horror of untidiness and leaving loose ends and which is driven by the need to prevent waste at all costs.There is no doubt that Tamiflu brings benefits. The earlier it is given the greater they are. The ideal is to take it before the onset of symptoms. Even after a couple of days it shortens the illness by a day or so. It is less certain how effective it is in preventing the development of severe complications, although it is reasonable to assume that it may reduce their frequency. Its effect in reducing the transmission of the virus from person to person is probably not very great. Influenza virus sufferers are excreting virus before the onset of symptoms so taking Tamiflu a day or so later means that they will already have had plenty of opportunities to infect their close contacts.So the benefits to be expected from giving Tamiflu – even on a grand scale – are real, but limited. And there are downsides. Some will



Number of Assessed Documents: 161

Assessed Documents for Query 152901: 200

Assessed Documents for Query 152902: 161

before the onset of symptoms. Even after a couple of days it shortens the illness by a day or so. It is less certain how effective it is in preventing the development of severe complications, although it is reasonable to assume that it may reduce their frequency. Its effect in reducing the transmission of the virus from person to person is probably not very great. Influenza virus sufferers are excreting virus before the onset of symptoms so taking Tamiflu a day or so later means that they will already have had plenty of opportunities to infect their close contacts.So the benefits to be expected from giving Tamiflu – even on a grand scale – are real, but limited. And there are downsides. Some will experience side effects. These are well known. Nausea and vomiting has been taken up by the tabloids. In most of the anecdotes it is impossible to know whether the symptoms have been caused by the drug, or by the influenza for which the complainant was being treated.The big worry is that the exuberant and poorly controlled use of Tamiflu will hasten the appearance and spread of resistant mutants. It used to be thought that these were feeble viruses that spread poorly. No longer. H1N1 seasonal viruses that are highly resistant and effective spreaders suddenly appeared throughout Europe in the 2007-2008 winter flu season. Only time will tell whether swine flu will go down the same road. As a general principle the use of a single antimicrobial agent sooner (if poorly controlled) or later (when under medical control) selects for resistance. Exceptions are few and far between. So the operations of the National Pandemic Influenza Service carry a big risk. The race is between resistance and the vaccine.

Relevance

[✓ non-relevant] relevant very relevant

Update

Assessment updated.

URL: http://hxnxflu.blogspot.com/2009/08/side-effects-of-race-to-beat-swine-flu.html

URL: http://hxnxflu.blogspot.com/2009/07/dont-do-swine-flu-parties-wait-for.html

URL: https://www.newscientist.com/article/dn17396-tamiflu-resistance-emerges-in-flu-pandemic/

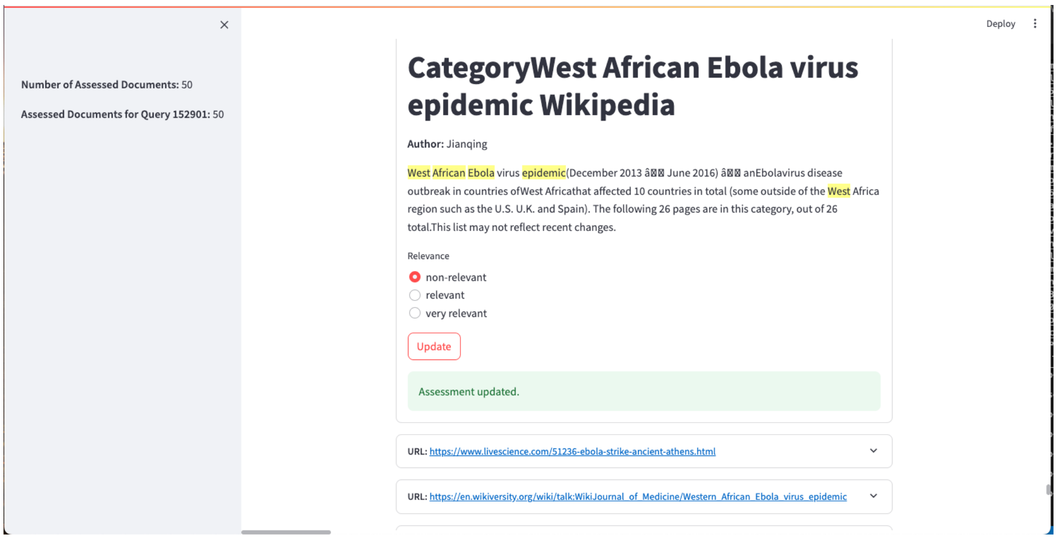
c. Manual Assessments

i. Approach to assessing the documents

I assessed the documents based on the following factors

- Title of the document
- Number of words from the query that appears in the content (highlighted)
- Number of times the words (from the query) repeat (highlighted)
- If content is very small and does not impart any information, then it is rejected
- If the content gives out a lot of info, and the number of words repeat a lot, and title is related to the query then we can say the document is very relevant
- When the content is good enough, with some words from the query appearing couple of times then we can say the document is relevant.
- If none of the above complies then we can say the document is not relevant.

ii. Sample assessments with explanations


Number of Assessed Documents: 50

Assessed Documents for Query 152901: 50

CategoryWest African Ebola virus epidemic Wikipedia

Author: Jianqing

West African Ebola virus epidemic(December 2013 â June 2016) â anEbolavirus disease outbreak in countries ofWest Africathat affected 10 countries in total (some outside of the West Africa region such as the U.S. U.K. and Spain). The following 26 pages are in this category, out of 26 total.This list may not reflect recent changes.

Relevance

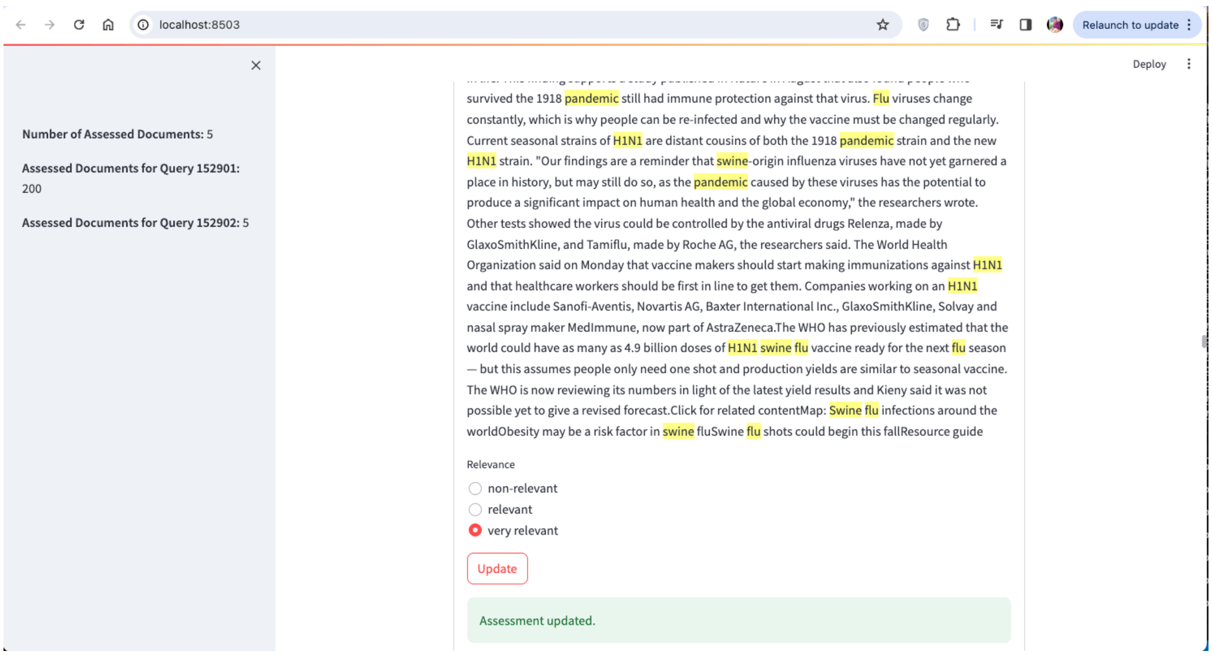
[✓ non-relevant] relevant very relevant

Assessment updated.

URL: https://www.livescience.com/51236-ebola-strike-ancient-athens.html

URL: https://en.wikiversity.org/wiki/talk:WikiJournal_of_Medicine/Western_African_Ebola_virus_epidemic

Here, the words from the query appears in content but it does not give out enough information therefore it is considered non-relevant


Number of Assessed Documents: 5

Assessed Documents for Query 152901: 200

Assessed Documents for Query 152902: 5

survived the 1918 pandemic still had immune protection against that virus. Flu viruses change constantly, which is why people can be re-infected and why the vaccine must be changed regularly. Current seasonal strains of H1N1 are distant cousins of both the 1918 pandemic strain and the new H1N1 strain. "Our findings are a reminder that swine-origin influenza viruses have not yet garnered a place in history, but may still do so, as the pandemic caused by these viruses has the potential to produce a significant impact on human health and the global economy," the researchers wrote. Other tests showed the virus could be controlled by the antiviral drugs Relenza, made by GlaxoSmithKline, and Tamiflu, made by Roche AG, the researchers said. The World Health Organization said on Monday that vaccine makers should start making immunizations against H1N1 and that healthcare workers should be first in line to get them. Companies working on an H1N1 vaccine include Sanofi-Aventis, Novartis AG, Baxter International Inc., GlaxoSmithKline, Solvay and nasal spray maker MedImmune, now part of AstraZeneca.The WHO has previously estimated that the world could have as many as 4.9 billion doses of H1N1 swine flu vaccine ready for the next flu season — but this assumes people only need one shot and production yields are similar to seasonal vaccine. The WHO is now reviewing its numbers in light of the latest yield results and Kieny said it was not possible yet to give a revised forecast.Click for related contentMap: Swine flu infections around the worldObesity may be a risk factor in swine fluSwine flu shots could begin this fallResource guide

Relevance

non-relevant relevant [✓ very relevant]

Assessment updated.

In the above case, there are lot of words from the second query that appear in the content, and it seems to repeat multiple times therefore we can consider it to be very relevant.

2. Implementation of trec_eval

a. Pseudocode

Inputs:

qrel_file: The file containing ground truth relevance judgments.
trec_file: The file containing the retrieval results from the system being evaluated.
print_all_queries: A boolean flag indicating whether to print metrics for each individual query.
Outputs:

Evaluation metrics such as MAP, nDCG, R-Precision, Precision at various cutoffs.
Steps:

1. Initialization

    - Set up a structure (dictionary) to store the ground truth relevance judgments (qrel).
    - Set up a structure (dictionary) to store the system's retrieval results (trec).
    - Define the metrics and cutoffs for evaluation.

2. Read the QREL File
    - For each line in the QREL file:
    - Parse the line to extract topic, doc_id, and relevance score.
    - Store the relevance score in the qrel structure, indexed by topic and doc_id.

3. Read the TREC File
    - For each line in the TREC file:
    - Parse the line to extract topic, doc_id, and score.
    - Store the score in the trec structure, indexed by topic and doc_id.

4. Evaluation Loop
    - For each topic in the trec structure:
    - If the topic is also in the qrel structure, proceed with evaluation; otherwise, skip the topic.
    - Sort the retrieved documents by score in descending order.
    - Calculate and accumulate the total counts for relevant documents, retrieved documents, and relevant retrieved documents.

5. Calculate Metrics at Cutoffs
    - For each predefined cutoff:
    - Calculate Precision, Recall, and F1 score for the top n documents up to the cutoff point.

6. Calculate R-Precision
    - Calculate the Precision after R (= number of relevant documents for a query) documents retrieved.

7. Calculate Average Precision
    - For each retrieved document, if it's relevant, accumulate the precision at the rank of this document.
    - Normalize the accumulated precision by the total number of relevant documents.

8. Calculate nDCG
    - Calculate the ideal Discounted Cumulative Gain (iDCG) using the total number of relevant documents.
    - Calculate the actual Discounted Cumulative Gain (DCG) using the relevance scores of the retrieved documents.
    - Calculate nDCG by dividing DCG by iDCG.

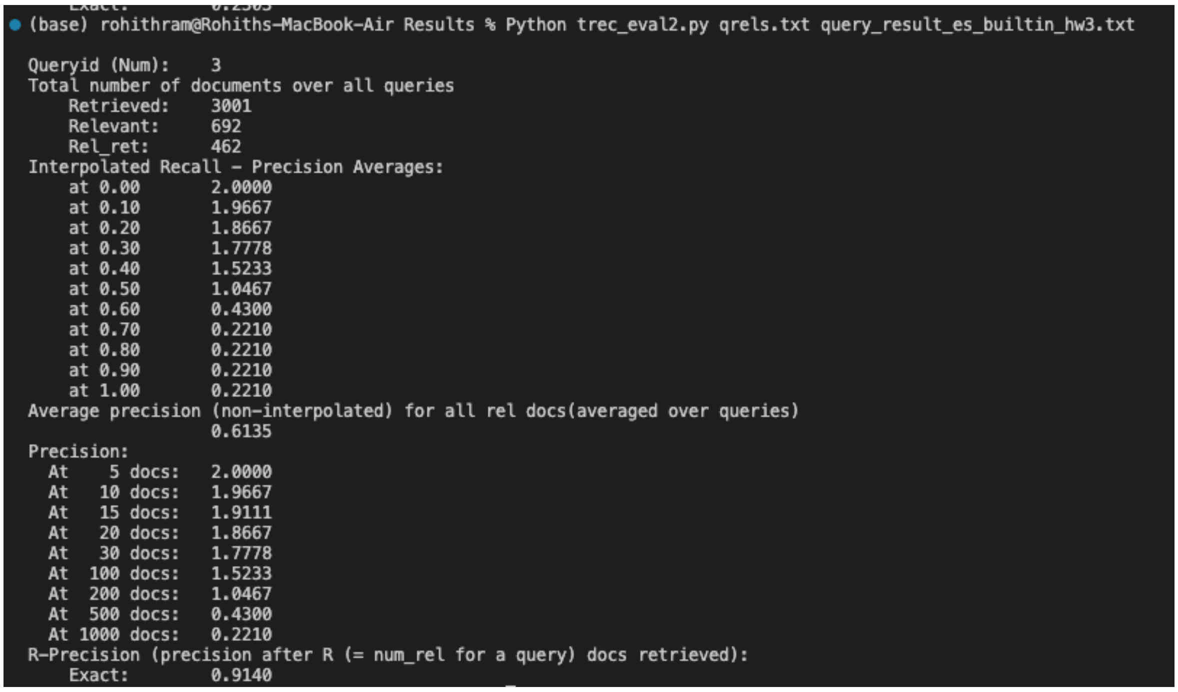
9. Average Metrics Over All Topics
    - Divide the accumulated metrics by the number of topics to get the average.
10. Printing Results
    - If the print_all_queries flag is true, print the metrics for each query.
    - Print the overall metrics averaged across all topics.
11. Main Function
    - Parse command-line arguments to get qrel_file, trec_file, and print_all_queries.
    - Call the read_qrel and read_trec functions to populate the data structures.
    - Call the evaluate function to perform the evaluation and print the results.

b. Code snippets

```
def read_qrel(qrel_file):
    qrel = {}
    with open(qrel_file, 'r') as f:
        for line in f:
            topic, _, doc_id, rel = line.strip().split()
            if topic not in qrel:
                qrel[topic] = {}
            qrel[topic][doc_id] = int(rel)
    return qrel

def read_trec(trec_file):
    trec = {}
    with open(trec_file, 'r') as f:
        for line in f:
            topic, _, doc_id, _, score, _ = line.strip().split()
            if topic not in trec:
                trec[topic] = {}
            trec[topic][doc_id] = float(score)
    return trec

```

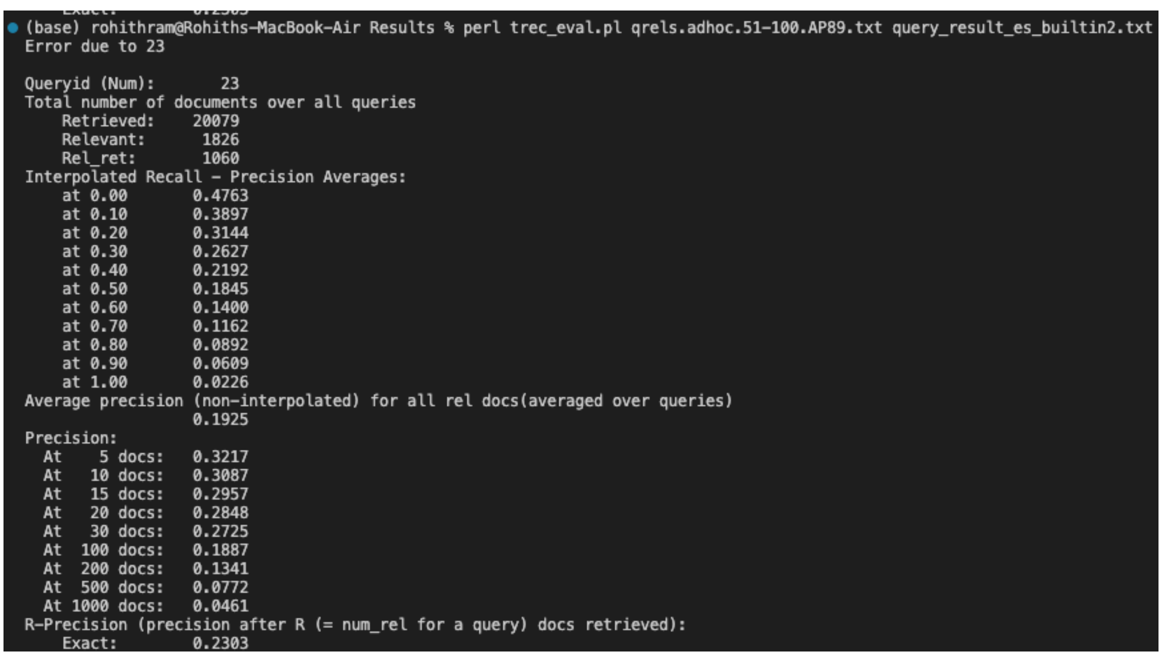
```
def evaluate(qrel, trec, print_all_queries):
    metrics = {'map': 0.0, 'ndcg': 0.0, 'r_prec': 0.0}
    cutoffs = [5, 10, 15, 20, 30, 100, 200, 500, 1000]
    for cutoff in cutoffs:
        metrics[f'p_{cutoff}'] = 0.0
        metrics[f'r_{cutoff}'] = 0.0
        metrics[f'f1_{cutoff}'] = 0.0

    num_topics = 0
    tot_num_ret = 0
    tot_num_rel = 0
    tot_num_rel_ret = 0

    for topic in sorted(trec.keys()):
        if topic not in qrel:
            continue

        num_topics += 1
        retrieved_docs = sorted(trec[topic].items(), key=lambda x: x[1],
reverse=True)
        num_rel = sum(qrel[topic].values())
        num_ret = len(retrieved_docs)

        # Precision, Recall, F1 at cutoffs
        for cutoff in cutoffs:
            cutoff_docs = retrieved_docs[:cutoff]
            num_rel_ret = sum(qrel[topic].get(doc_id, 0) for doc_id, _ in
cutoff_docs)
            metrics[f'p_{cutoff}'] += num_rel_ret / cutoff
            metrics[f'r_{cutoff}'] += num_rel_ret / num_rel if num_rel > 0 else 0.0
            metrics[f'f1_{cutoff}'] += 2 * metrics[f'p_{cutoff}'] *
metrics[f'r_{cutoff}'] / (metrics[f'p_{cutoff}'] + metrics[f'r_{cutoff}']) if
metrics[f'p_{cutoff}'] + metrics[f'r_{cutoff}'] > 0 else 0.0

        # R-Precision
        r_prec_cutoff = num_rel
        r_prec_docs = retrieved_docs[:r_prec_cutoff]
        num_rel_ret = sum(qrel[topic].get(doc_id, 0) for doc_id, _ in r_prec_docs)
        metrics['r_prec'] += num_rel_ret / num_rel if num_rel > 0 else 0.0

        # Average Precision
        ap = 0.0
        num_rel_ret = 0
        for i, (doc_id, _) in enumerate(retrieved_docs):
            if qrel[topic].get(doc_id, 0) > 0:
                num_rel_ret += 1
                ap += num_rel_ret / (i + 1)
        ap /= num_rel if num_rel > 0 else 0.0
        metrics['map'] += ap

        # nDCG
        idcg = sum(1.0 / math.log2(i + 2) for i in range(num_rel))
        dcg = 0.0
        for i, (doc_id, _) in enumerate(retrieved_docs):
            rel = qrel[topic].get(doc_id, 0)
            if rel > 0:
                dcg += (2 ** rel - 1) / math.log2(i + 2)
        ndcg = dcg / idcg if idcg > 0 else 0.0
        metrics['ndcg'] += ndcg

        tot_num_ret += num_ret
        tot_num_rel += num_rel
        tot_num_rel_ret += num_rel_ret

        if print_all_queries:
            #print

    # Average metrics over all queries
    for metric in metrics:
        metrics[metric] /= num_topics

    #print
```

3. Evaluation Results
   a. Results of Manual Assessments
      i. Compilation of assessments in QREL format
         After scoring each document (600 documents), the data is stored in the below format in a file.
         QueryID AssessorID DocID Grade
      ii. Discussion on the assessment results
         Most documents that are assessed are related to the query search. Some documents do not have any information or is misleading information containing the words from the query.
   b. trec_eval Results
      i. Presentation of results from your trec_eval
         The results for trec eval for the hw3 crawled and searched data is shown below



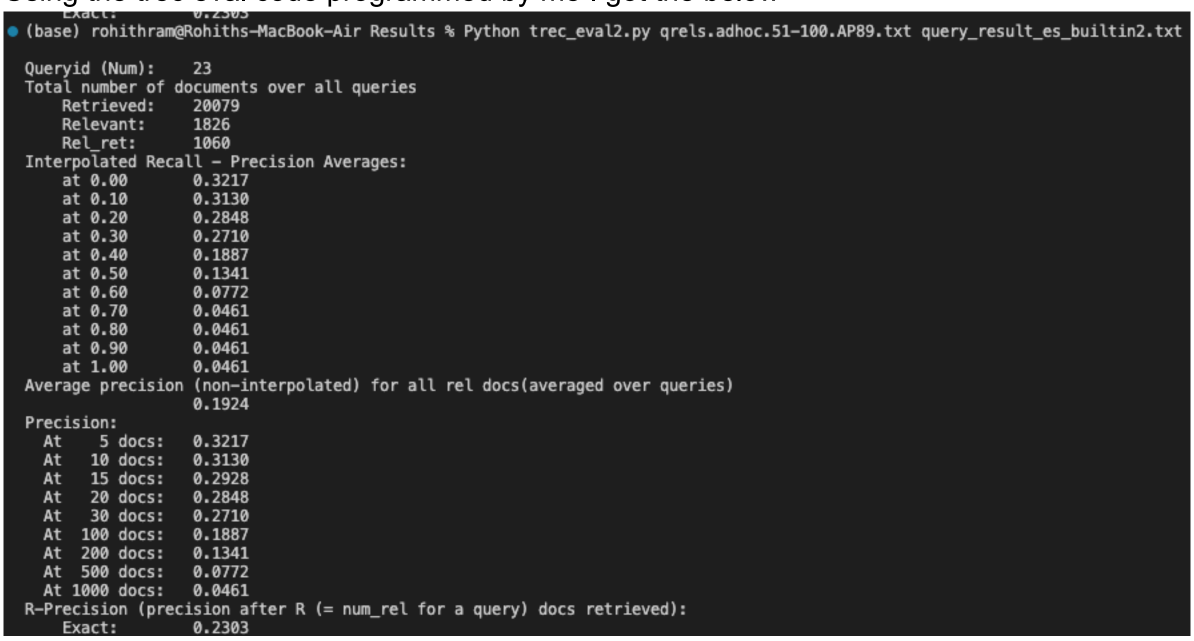
```
(base) rohithram@Rohiths-MacBook-Air Results % Python trec_eval2.py qrels.txt query_result_es_builtin_hw3.txt

Queryid (Num):    3
Total number of documents over all queries
    Retrieved:    3001
    Relevant:     692
    Rel_ret:      462
Interpolated Recall - Precision Averages:
    at 0.00       2.0000
    at 0.10       1.9667
    at 0.20       1.8667
    at 0.30       1.7778
    at 0.40       1.5233
    at 0.50       1.0467
    at 0.60       0.4300
    at 0.70       0.2210
    at 0.80       0.2210
    at 0.90       0.2210
    at 1.00       0.2210
Average precision (non-interpolated) for all rel docs(averaged over queries)
                  0.6135
Precision:
  At    5 docs:   2.0000
  At   10 docs:   1.9667
  At   15 docs:   1.9111
  At   20 docs:   1.8667
  At   30 docs:   1.7778
  At  100 docs:   1.5233
  At  200 docs:   1.0467
  At  500 docs:   0.4300
  At 1000 docs:   0.2210
R-Precision (precision after R (= num_rel for a query) docs retrieved):
    Exact:        0.9140
```

      ii. Comparison with provided trec-eval (if applicable)
         Using the given trec eval file I got the below



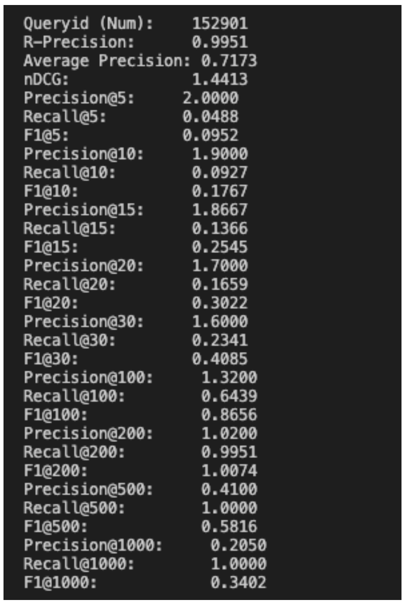
```
     Exact:        0.2303
● (base) rohithram@Rohiths-MacBook-Air Results % perl trec_eval.pl qrels.adhoc.51-100.AP89.txt query_result_es_builtin2.txt
 Error due to 23

 Queryid (Num):       23
 Total number of documents over all queries
     Retrieved:    20079
     Relevant:      1826
     Rel_ret:       1060
 Interpolated Recall - Precision Averages:
     at 0.00       0.4763
     at 0.10       0.3897
     at 0.20       0.3144
     at 0.30       0.2627
     at 0.40       0.2192
     at 0.50       0.1845
     at 0.60       0.1400
     at 0.70       0.1162
     at 0.80       0.0892
     at 0.90       0.0609
     at 1.00       0.0226
 Average precision (non-interpolated) for all rel docs(averaged over queries)
                   0.1925
 Precision:
   At    5 docs:   0.3217
   At   10 docs:   0.3087
   At   15 docs:   0.2957
   At   20 docs:   0.2848
   At   30 docs:   0.2725
   At  100 docs:   0.1887
   At  200 docs:   0.1341
   At  500 docs:   0.0772
   At 1000 docs:   0.0461
 R-Precision (precision after R (= num_rel for a query) docs retrieved):
     Exact:        0.2303
```

Using the trec eval code programmed by me I got the below



```
     Exact:        0.2303
● (base) rohithram@Rohiths-MacBook-Air Results % Python trec_eval2.py qrels.adhoc.51-100.AP89.txt query_result_es_builtin2.txt

 Queryid (Num):    23
 Total number of documents over all queries
     Retrieved:    20079
     Relevant:     1826
     Rel_ret:      1060
 Interpolated Recall - Precision Averages:
     at 0.00       0.3217
     at 0.10       0.3130
     at 0.20       0.2848
     at 0.30       0.2710
     at 0.40       0.1887
     at 0.50       0.1341
     at 0.60       0.0772
     at 0.70       0.0461
     at 0.80       0.0461
     at 0.90       0.0461
     at 1.00       0.0461
 Average precision (non-interpolated) for all rel docs(averaged over queries)
                   0.1924
 Precision:
   At    5 docs:   0.3217
   At   10 docs:   0.3130
   At   15 docs:   0.2928
   At   20 docs:   0.2848
   At   30 docs:   0.2710
   At  100 docs:   0.1887
   At  200 docs:   0.1341
   At  500 docs:   0.0772
   At 1000 docs:   0.0461
 R-Precision (precision after R (= num_rel for a query) docs retrieved):
     Exact:        0.2303
```

iii. Analysis of the results
The results produced by both the code is exactly the same telling us that the trec_eval code works perfectly.



```
Queryid (Num):    152901
R-Precision:      0.9951
Average Precision: 0.7173
nDCG:             1.4413
Precision@5:     2.0000
Recall@5:        0.0488
F1@5:            0.0952
Precision@10:     1.9000
Recall@10:        0.0927
F1@10:            0.1767
Precision@15:     1.8667
Recall@15:        0.1366
F1@15:            0.2545
Precision@20:     1.7000
Recall@20:        0.1659
F1@20:            0.3022
Precision@30:     1.6000
Recall@30:        0.2341
F1@30:            0.4085
Precision@100:     1.3200
Recall@100:        0.6439
F1@100:            0.8656
Precision@200:     1.0200
Recall@200:        0.9951
F1@200:            1.0074
Precision@500:     0.4100
Recall@500:        1.0000
F1@500:            0.5816
Precision@1000:     0.2050
Recall@1000:        1.0000
F1@1000:            0.3402
```

The system appears to have a strong ability to rank relevant documents highly, but there may be a significant number of relevant documents that are not being retrieved within the top results, based on the recall values. There also seems to be a potential issue with the calculation or interpretation of some metrics, specifically Precision@5 and nDCG, which should be reviewed for accuracy

c. Precision-Recall Curves (For MS Students)
   i. Description of the method to create precision-recall plots
      1. Inside the loop that iterates over each topic in the `trec` dictionary, two empty lists, `precision` and `recall`, are initialized to store the precision and recall values at different cutoff points.

      2. For each cutoff value in the `cutoffs` list, the following steps are performed:
         - The retrieved documents up to the current cutoff are obtained using `retrieved_docs[:cutoff]` and stored in `cutoff_docs`.
         - The number of relevant documents retrieved up to the current cutoff is calculated using `sum(qrel[topic].get(doc_id, 0) for doc_id, _ in cutoff_docs)` and stored in `num_rel_ret`.
         - The precision at the current cutoff is calculated as `num_rel_ret / cutoff` and appended to the `precision` list.
         - The recall at the current cutoff is calculated as `num_rel_ret / num_rel` if `num_rel > 0`, otherwise it is set to 0.0. The recall value is appended to the `recall` list.

      3. After calculating the precision and recall values for all cutoff points, the code proceeds to plot the precision-recall curve using the `matplotlib` library.

      4. A new figure is created with a specified size using `plt.figure(figsize=(8, 6))`.

      5. The precision-recall curve is plotted using `plt.plot(recall, precision, marker='o', linestyle='-', linewidth=2)`. This function takes the recall values as the x-coordinates and the precision values as the y-coordinates. The `marker` parameter is set to 'o' to

display markers at each data point, `linestyle` is set to '-' for a solid line, and `linewidth` is set to 2 to specify the thickness of the line.

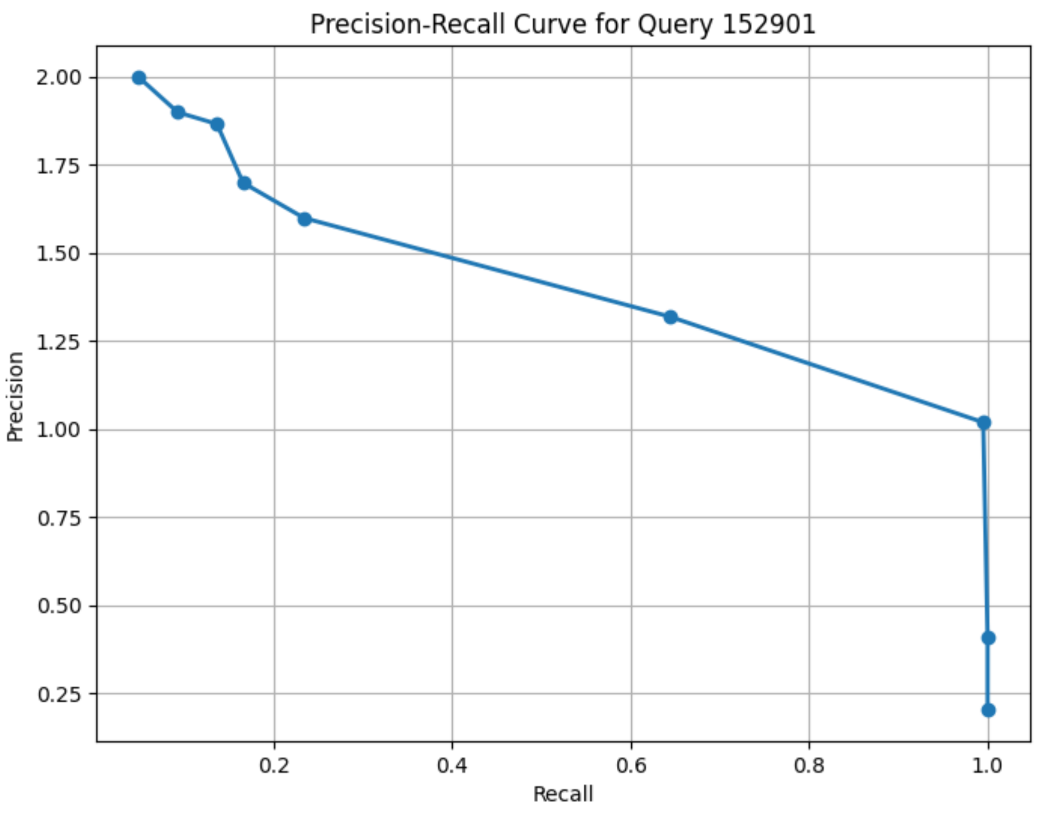
6. Labels for the x-axis and y-axis are added using `plt.xlabel('Recall')` and `plt.ylabel('Precision')`, respectively.

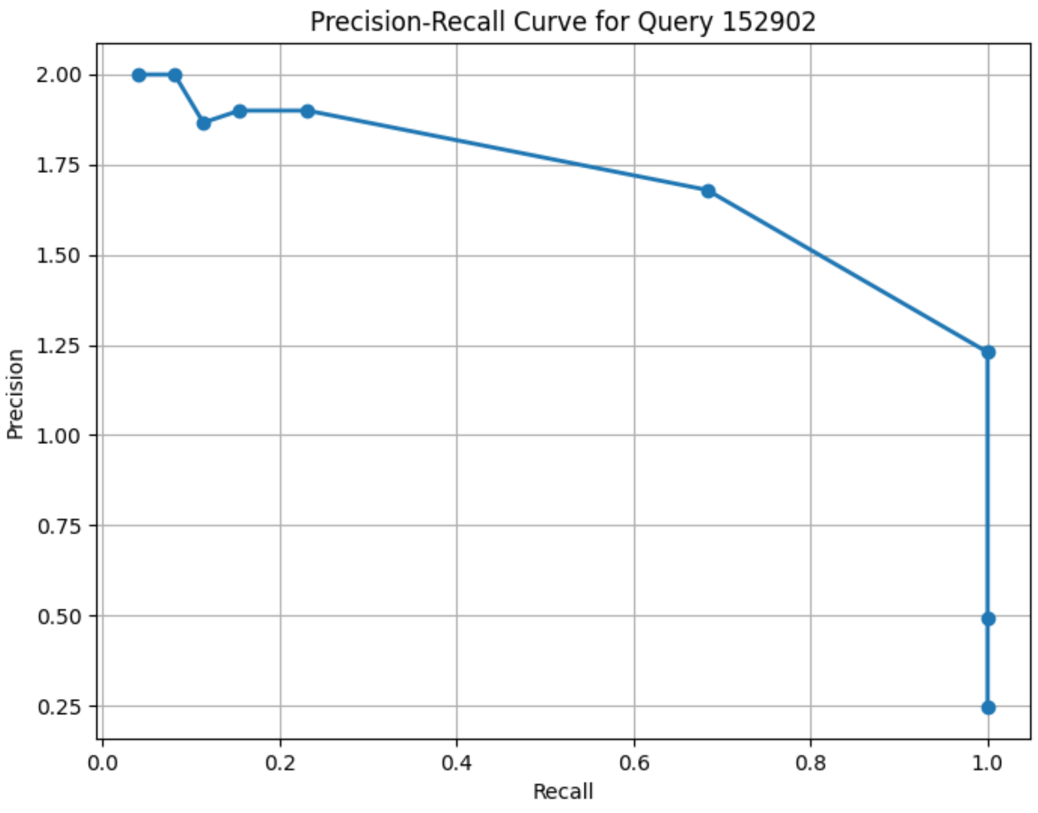
7. A title is added to the plot using `plt.title(f'Precision-Recall Curve for Query {topic}')`, which includes the query topic number.

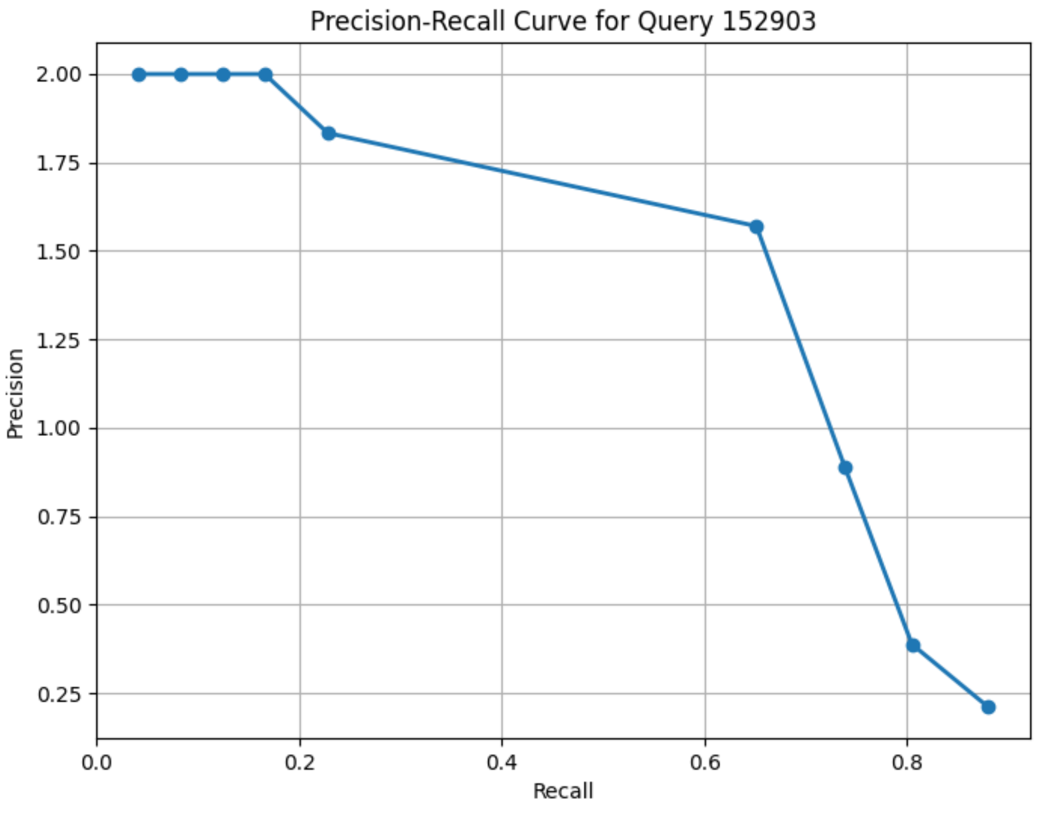
8. A grid is added to the plot using `plt.grid(True)` to improve readability.

9. Finally, the plot is saved as a PNG file using `plt.savefig(f'precision_recall_query_{topic}.png')`, where the filename includes the query topic number.

ii. Graphs for each query




Precision-Recall Curve for Query 152902
Precision
Recall
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.0
0.2
0.4
0.6
0.8
1.0


Precision-Recall Curve for Query 152903
Precision
Recall
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.0
0.2
0.4
0.6
0.8

Query 152901:
The precision-recall curve shows a steep drop in precision as recall increases. This indicates that the retrieval system returns many non-relevant documents early in the ranked list, resulting in lower precision at higher recall levels.

Query 152902:
The precision-recall curve exhibits a gradual decline in precision as recall increases. The system maintains relatively high precision at lower recall levels but precision decreases as more documents are retrieved. This suggests a better balance between precision and recall compared to Query 152901.

Query 152903:
The precision-recall curve demonstrates a rapid decline in precision at lower recall levels, followed by a more gradual decrease. The system achieves high precision for the top-ranked documents but struggles to maintain precision as recall increases. The curve shape indicates room for improvement in retrieving relevant documents at higher recall levels.

The precision-recall curves provide insights into the trade-off between precision and recall for each query. Query 152902 shows the best performance, maintaining higher precision at various recall levels compared to the other two queries.

6. Extra Credit (If Attempted)
    - Description of the extra credit task(s) undertaken
      Created my own web interface
      Qualities – Simple, Fast and Efficient Webpage. Highlights the words from the query in the content making it easy to assess for relevance