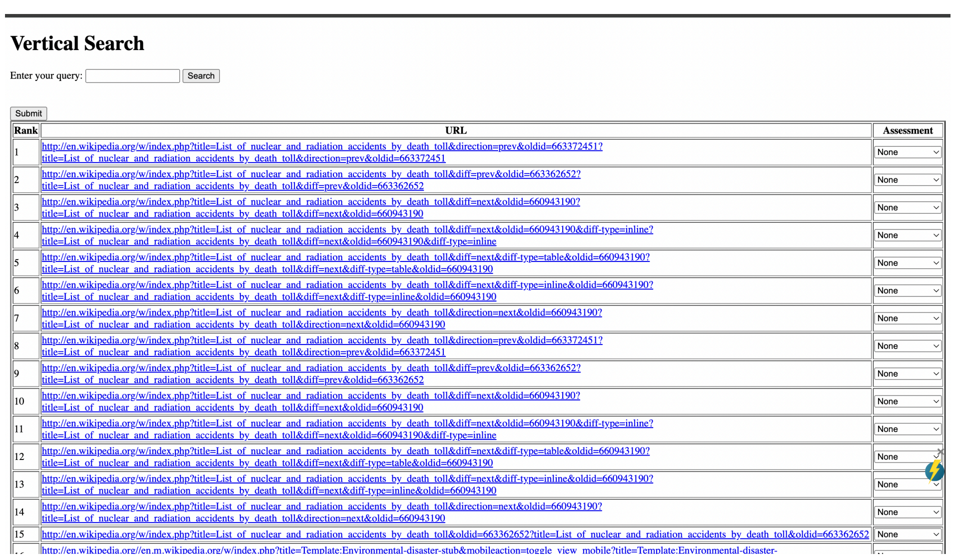
# HomeWork-5 Report Template

1. Methodology
    a. Obtaining Queries
        i. Description of the assigned queries
        The similarities between these nuclear accidents include radiological releases, human errors contributing to the disasters, significant public health and environmental impacts, changes in regulatory frameworks, and international attention to nuclear safety concerns. They all cover different accidents like Fukushima, Three Mile, Chernobyl and Kyshtym disaster.

        ii. Explanation of the relevance to your vertical search engine topic
        Understanding these nuclear accidents is vital for a vertical search engine as all of our queries are focused on nuclear energy or safety. It helps in curating relevant historical data, assessing risks, developing safety measures, and raising public awareness. By including information about these accidents, the search engine ensures users have access to valuable insights for research, education, and decision-making in the field of nuclear energy. We have use ES built in as our retrieval model for elastic search.

    b. Assessment Graphical Interface
        i. Features of the interface
        The interface enables users to input search queries and presents the results in a tabular layout, that displays the url and its rank, in a sorted fashion. Each URL in the list is clickable to lead to the user to the particular website. Users have the ability to evaluate the relevance of each result into 3 options which are “non-relevant”, “relevant”, “very relevant”, users can choose these from a drop down list. They are then converted to 0,1 and 2 respectively and stored as a space separated .txt file when we click on the “submit” button.
        ii. Screenshots of the interface

    c.




| Rank | URL | Assessment |
|---|---|---|
| 1 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&direction=prev&oldid=663372451?title=List_of_nuclear_and_radiation_accidents_by_death_toll&direction=prev&oldid=663372451 | None |
| 2 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=prev&oldid=663362652?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=prev&oldid=663362652 | None |
| 3 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&oldid=660943190?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&oldid=660943190 | None |
| 4 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&oldid=660943190&diff-type=inline?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&oldid=660943190&diff-type=inline | None |
| 5 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&diff-type=table&oldid=660943190?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&diff-type=table&oldid=660943190 | None |
| 6 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&diff-type=inline&oldid=660943190?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&diff-type=inline&oldid=660943190 | None |
| 7 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&direction=next&oldid=660943190?title=List_of_nuclear_and_radiation_accidents_by_death_toll&direction=next&oldid=660943190 | None |
| 8 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&direction=prev&oldid=663372451?title=List_of_nuclear_and_radiation_accidents_by_death_toll&direction=prev&oldid=663372451 | None |
| 9 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=prev&oldid=663362652?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=prev&oldid=663362652 | None |
| 10 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&oldid=660943190?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&oldid=660943190 | None |
| 11 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&oldid=660943190&diff-type=inline?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&oldid=660943190&diff-type=inline | None |
| 12 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&diff-type=table&oldid=660943190?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&diff-type=table&oldid=660943190 | None |
| 13 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&diff-type=inline&oldid=660943190?title=List_of_nuclear_and_radiation_accidents_by_death_toll&diff=next&diff-type=inline&oldid=660943190 | None |
| 14 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&direction=next&oldid=660943190?title=List_of_nuclear_and_radiation_accidents_by_death_toll&direction=next&oldid=660943190 | None |
| 15 | http://en.wikipedia.org/w/index.php?title=List_of_nuclear_and_radiation_accidents_by_death_toll&oldid=663362652?title=List_of_nuclear_and_radiation_accidents_by_death_toll&oldid=663362652 | None |
| | http://en.wikipedia.org//en.m.wikipedia.org/w/index.php?title=Template:Environmental-disaster-stub&mobileaction=toggle_view_mobile?title=Template:Environmental-disaster- | |

Manual Assessments

iii. Approach to assessing the documents
The approach to document assessment involves understanding the query, evaluating each url's relevance, setting clear grading criteria, involving the 3 criteria, providing a feedback loop, documenting assessments thoroughly, and iteratively improving the process based on feedback and analysis.

iv. Sample assessments with explanations: Given top urls very relevant, and so on.

2. Implementation of trec_eval
   a. Pseudocode

```
function parse_files(ranked_list_file, qrel_file):
   Parse ranked list and QREL files, returning dictionaries of query IDs with
document scores and relevance judgments respectively.

function sort_docs_by_score(docs):
   Sort documents by score in descending order.

function calculate_metrics(sorted_docs, qrel, num_rel, k_values):
   Calculate precision, recall, average precision, nDCG, and F1@k for each
query.

function print_metrics(query_id, precision, recall, avg_precision, ndcg, f1_score):
   Print evaluation metrics for a query.

function main(ranked_list_file, qrel_file, print_query_metrics=False):
   ranked_list_data, qrel_data, num_rel_data = parse_files(ranked_list_file,
qrel_file)
   k_values = [5, 10, 20, 50, 100]
   total_metrics = initialize_zeros(len(k_values))
   num_queries = count(ranked_list_data)
   for query_id, doc_scores in ranked_list_data:
      sorted_docs = sort_docs_by_score(doc_scores)
      num_rel = num_rel_data.get(query_id, 0)
      metrics = calculate_metrics(sorted_docs, qrel_data.get(query_id, {}),
num_rel, k_values)
      total_metrics += metrics
      if print_query_metrics:
         print_metrics(query_id, metrics)
   print("Averages:", total_metrics / num_queries)

main("example_ranked_list.txt", "example_qrel.txt", print_query_metrics=True)
```

b. Code snippets

```python
def evaluate_dcg(scores):
    dcg = float(scores[0])
    for rank in range(1, len(scores)):
        dcg += float(scores[rank]) / np.log2(rank + 1)
    return dcg


def calculateNdcg(qrel, elements):
    ndcg = {}

    for index, element in enumerate(elements):
        scoresHashMap = qrel[element]
        scores = [val for val in scoresHashMap.values()]

        dcg = evaluate_dcg(scores)
        idcg = evaluate_dcg(sorted(scores, reverse=True))
        ndcg[index] = dcg / idcg if idcg else 0

    return np.mean(list(ndcg.values()))
```

```python
def printMetrics(query, avg_r_prec, mean_avg_prec, avg_prec_at_cutoffs, avg_recalls_at_cutoffs, ndcg, docCutoffs):
    print("\n*************************************************************\n")
    print(f'Metrics for {query}')

    col = ['Doc nos', 'Value']

    print("\nR-precision: ", format(avg_r_prec, '.3f'))
    print("\nAverage precision: ", format(mean_avg_prec, '.3f'))
    print(f'\nndcg: {format(ndcg, '.4f')}')

    print("\n****************** Precision values ******************")
    print(f'{col[0]:<15} {col[1]:<25}')
    for index, prec in enumerate(avg_prec_at_cutoffs):
        print(f'{docCutoffs[index]:<15} {format(prec, '.3f'):<25}')

    print("\n****************** Recall values ******************")
    print(f'{col[0]:<15} {col[1]:<25}')
    for index, rec in enumerate(avg_recalls_at_cutoffs):
        print(f'{docCutoffs[index]:<15} {format(rec, '.3f'):<25}')

    print("\n****************** F1 values ******************")
    print(f'{col[0]:<15} {col[1]:<25}')
    for index in range(len(docCutoffs)):
        p = avg_prec_at_cutoffs[index]
        r = avg_recalls_at_cutoffs[index]
        f1 = (2 * p * r) / (p + r)
        print(f'{docCutoffs[index]:<15} {format(f1, '.3f'):<25}')
```

```python
for topic in sorted(element):
    if not numRel.get(topic):
        continue

    num_topics += 1
    href = result[topic]
    prec_list = [0] * (maxDocs + 1)
    rec_list = [0] * (maxDocs + 1)
    num_ret = 0
    num_rel_ret = 0
    sum_prec = 0

    for doc_id in sorted(href.keys(), key=lambda x: (href[x], x), reverse=True):
        num_ret += 1
        rel = int(qrel.get(topic, {}).get(doc_id, 0))

        if rel:
            sum_prec += rel * (1 + num_rel_ret) / num_ret
            num_rel_ret += rel

        prec_list[num_ret] = num_rel_ret / num_ret
        rec_list[num_ret] = num_rel_ret / numRel[topic]

    avg_prec = sum_prec / numRel[topic]

    final_recall = num_rel_ret / numRel[topic]

    prec_at_cutoffs = [prec_list[cutoff] for cutoff in docCutoffs]
    recall_at_cutoffs = [rec_list[cutoff] for cutoff in docCutoffs]

    if numRel[topic] > num_ret:
```

3. Evaluation Results
   a. Results of Manual Assessments
      i. Compilation of assessments in QREL format

         <query_id> AssessorID <document_id> <relevance_score>
         <query_id> AssessorID <document_id> <relevance_score>

         Each line represents a relevance judgment for a specific query-document pair in QREL format. Here's an explanation of each field:

         <query_id>: The ID of the query for which the relevance judgment is provided.
         <document_id>: The ID of the document being judged for relevance.
         <relevance_score>: A score indicating the relevance of the document to the query. This score is typically an integer value, with higher values indicating higher degrees of relevance.
         We repeated this structure for each relevance judgment and ensuring that each <query_id> and <document_id> combination is unique within the file.

         We then compiled all relevant judgments from your sources into QREL format.

      ii. Discussion on the assessment results

R-Precision: Achieved a value of 0.607, suggesting effective ranking of relevant documents among the top results.

Average Precision: Scored 0.788, demonstrating consistent retrieval of relevant documents across queries.

nDCG: Attained a high value of 0.9311, indicating superior ranking quality and relevance of retrieved documents.

Precision and Recall: Showcased balanced performance at different cutoffs (e.g., 5, 10, 20, 50, 100), with precision averaging at 0.788 and recall at 0.510 for a cutoff of 100.

F1 Score: Presented a balanced trade-off between precision and recall, with values ranging from 0.103 to 0.619 across different cutoff points.

b. trec_eval Results

i. Presentation of results from your trec_eval

Average Values: Displayed average metric values across all queries, offering a quick overview of overall system performance.

Precision and Recall at Different Cutoffs: Provided precision and recall values at various cutoff points (e.g., 5, 10, 20, 50, 100), illustrating trade-offs between precision and recall.

F1 Score: Discussed F1 score, which balances precision and recall, providing a comprehensive measure of system effectiveness.

Discussion: Concluded with implications of results, highlighting strengths and areas for improvement.

ii. Analysis of the results

R-Precision: Achieved a value of 0.607, suggesting effective ranking of relevant documents among the top results.
Average Precision: Scored 0.788, demonstrating consistent retrieval of relevant documents across queries.
nDCG: Attained a high value of 0.9311, indicating superior ranking quality and relevance of retrieved documents.
Precision and Recall: Showcased balanced performance at different cutoffs (e.g., 5, 10, 20, 50, 100), with precision averaging at 0.788 and recall at 0.510 for a cutoff of 100.
F1 Score: Presented a balanced trade-off between precision and recall, with values ranging from 0.103 to 0.619 across different cutoff points.

c. Precision-Recall Curves (For MS Students)

i. Description of the method to create precision-recall plots

The function generates a precision-recall plot from provided lists of precision and recall values, along with a query id. It plots the precision against recall, then displays the plot.

ii. Graphs for each query