UNIVERSITY OF SOUTHERN MAINE

Introduction to Information Retrieval, Fall 2024, Assignment 2

Instructor: Behrooz Mansouri

Due: October 08, 2024

In this assignment, you will explore more advanced information retrieval models; TF-IDF (system 1) and BM25 (system 2) in Python. The goal is to find relevant answers to given questions about traveling.

Data files

The data files remain similar to the first assignment.

Designing Retrieval Models

You will implement your own information retrieval models (not from other Python libraries and existing search engines). You will implement both TF-IDF and BM25 models. You can design how to implement these models (for optimal efficiency and effectiveness); but here are two suggestions:

1. **Vector Space Model**: represent all the documents as vectors of words (unique tokens in your collection). The values in a vector show the TF-IDF/BM25 values for that token in the corresponding document. For search, you will convert your input question to a similar vector and you can simply use cosine similarity.

2. **Inverted Index**: consider a dictionary of dictionaries. The keys in the first dictionary are tokens and the values are in another dictionary. In this dictionary, keys are the document IDs and values are the (token, document) TF-IDF/BM25 values. For search, you will get all the terms in the questions and calculate TF-IDF/BM25 scores.

Evaluation Measures

Using the qrel_1 file, you should provide evaluation results for topics in topic_1 file. You will report the P@1, P@5, nDCG@5, MRR, MAP. For metrics other than nDCG, the relevant documents are those with scores 1 and 2 (in the qrel file). You can use any tool that you want for evaluation. However, this tool should be standard, and the results should be reproducible (Trec_eval and Ranx are recommended).

Report and Analysis

You should provide four results files; you have two systems (TF-IDF and BM25), and two inputs (topic_1 and topic_2). Make sure the result files can be passed to Trec_eval and Ranx for evaluation with the correct format. You do not need to provide an evaluation script.
For topic_1 file, provide a table showing the values for each metric (average over all the topics). Then provide a ski-jump plot based on P@5. For this plot, discuss what your models worked for, and what are the reasons for failures. You should provide at least one pair of successful and failed (topic, answer) in your discussion for each model.

Notes for submission:

1. Python file/files (only .py is acceptable, not .ipynb) with codes for all retrieval. Codes should be well-structured with comments to run. You should have a README file that provides clear guidance on how to run your code and get the result files. Explicit file path should not be used in the code, and topic files should be passed as arguments to your code.
2. result_tfidf_1.tsv and result_tfidf_2.tsv, result_bm25_1.tsv and result_bm25_2.tsv files having your retrieval result files. The _2 files correspond to the results for topic_2.
3. A .pdf file with your results table, ski-jump plot, and analysis
4. Any assumptions made by students should be explicitly mentioned in the submitted

Note: Any submission not in the format explained above will be dropped, resulting in 0, without the possibility of regrading