CENG 596 Information Retrieval
Spring 2024

Course Project

The objective of this course project is to provide you with practical, hands-on experience in addressing practical information retrieval challenges. You will be focusing on constructing an ad-hoc information retrieval system. This system should incorporate selected advanced features (of your choice), and its performance will be assessed using evaluation metrics widely used in the field.

General steps

1. Pick a data set:
   - You can find some interesting datasets at:
     https://ir-datasets.com/
     Make sure your dataset includes documents, queries and relevance judgements.
   - Local datasets:
     https://drive.google.com/drive/folders/1QWpclDXhw2OnRXjzMtPBJfqdnpypY_l0?usp=sharing
     The Google drive link provided by the instructor includes Washington Post dataset, Associated Press dataset, and Financial Times dataset.
     Terms of use: All datasets provided by the instructor are to be exclusively used for Ceng596 project only.

2. Based on the selected dataset (or subset of it) propose an information retrieval project using the techniques you learned in class. These may include but not limited to:
   - Make use of stemming, elimination of stop words, or any other preprocessing
   - Phrase queries, positional queries
   - Query expansion
   - Relevance feedback
   - Tolerant retrieval
   - Extended Boolean model
   - Vector space model
   - Probabilistic models
   - Language models
   - Learning to rank
   - Neural ranking

3. Over this collection, use **Lucene** or **PyTerrier** to create an inverted index. Then, process the queries over the inverted index testing several different ranking techniques and querying features as you stated in your proposal. Feel free to add any optimizations or components that could improve the evaluation scores. (If you want to experiment with neural ranking, you will need to create a dense index of document embeddings using large language models like BERT or GPT)

4. Evaluate the performance of different ranking schemes to conclude the best implementation alternative (You may use BM25 scoring function as a baseline). You can compare and combine results from different ranking schemes to come up with your own ranking scheme for the puposes of your project. To evaluate the performance, you can use the existing

evaluation tools (such as trec_eval etc.) by arranging your output format as required by these tools. (You can use the trec_eval script, the latest version)
.

5. Documentation: You will write a proposal, progress report and a final report. The templates for the reports will be provided.

6. Presentation of the project: Each project team is expected to present their project at the end of the semester. A demo of your implementation and GitHub codebase will be required.

Project Milestones:

- Project proposal (15%) 1 page . Due April 14.
- Project progress report (15%) 1-2 pages Due May 12.
- Presentation and demo, GitHub codebase (50%) Due June 7
  - 15 minutes presentation about what you have done for this course project. Format could be tailored according to the nature of the project, e.g., slides presentation, poster and/or system demo.
  - You will also create a repository for your project and upload your code on GitHub. Please ensure that your code is well-documented and that you have included a README file that explains how to run your code
- Project final report (20%) Due June 7
  - Detailed written report of your project.
  - Similar to a research paper, i.e., in formal written English and conference paper format.

References

1. Jimmy Lin, Ronak Pradeep, Tommaso Teofili, Jasper Xian: Vector Search with OpenAI Embeddings: Lucene Is All You Need. CoRR abs/2308.14963 (2023)
2. https://huggingface.co/blog/getting-started-with-embeddings
3. https://www.sbert.net/examples/applications/retrieve_rerank/README.html
4. https://www.sbert.net/docs/pretrained_models.html
5. Craig Macdonald, Nicola Tonellotto, Sean MacAvaney, Iadh Ounis: PyTerrier: Declarative Experimentation in Python from BM25 to Dense Retrieval. CIKM 2021: 4526-4533
6. https://pyterrier.readthedocs.io/en/latest/neural.html