Information Retrieval Project 2023-2024

Prof. Toon Calders, Ewoenam Tokpo
{toon.calders, ewoenamkwaku.tokpo}@uantwerpen.be

October 16, 2023

Deadlines

- proposal **30/10/2023**;
- final report: **12/12/2023**;
- slides: before presentation;
- presentation: week of **18/12/2023**

For the report see the guidelines below. For the proposal we expect the following information:

- **Admin:** Group members (2 or 3): names and IDs.
- **Scope:** What do you want to do. This item will allow us to give you timely feedback on the level of ambition.
- **Resources:** Data sources and tools you plan to use. Are there certain libraries you will explore an might use?
- **Literature:** External sources you plan to read.
- **Evaluation:** How will you test success? E.g. how will you get labels of what are the correct answers? Which datasets will you use?

You will get feedback on your proposal the latest on 13/11/2023. If you have questions after receiving the feedback, please contact us.

Overview

For further inquiries about the project, please email *ewoenamkwaku.tokpo@uantwerpen.be*.

- This project is to be executed in groups of 2 or 3 students. Please contact us if you are unable to get a project partner.
- We give three project examples that cover key concepts discussed in class. You are allowed to pick one of these projects. However, you are encouraged to propose your own project which matches the scale and the essence of the examples given — Recognition will be given to original projects. The project can be carried out with any programming language of choice.

- Each group will have to prepare a report, a demo, and a presentation to present their project to the other groups after the project has been completed — see **??**. This is a helpful exercise since you may be working on different projects related to different aspects of the course. More information regarding the schedule of the presentation will be given later.

1 Project

For the project, you are encouraged to pick any interesting information retrieval problem or task that can be carried out with the techniques explored in class. The entire project should incorporate *at least 3 key concepts* discussed in class.

1.1 Project examples

1. **Query-based Multi-document Summarizing:** Build a system that collects and collates relevant information to a query from multiple documents in an archive or database into a single document. Given a query, the retrieval system should retrieve all relevant documents, extract relevant information from each retrieved document, and summarize these pieces of information into a single document. The wikiHow dataset can be used to evaluate your implementation. The dataset comes with ground-truth labels that can be used to quantitatively evaluate your work.

2. **Fact Verification:** Implement a claim verification system that, given a claim, predicts whether the claim is *Factual*, *Untrue*, or *Unsubstantiated*. The system should verify a claim using multiple textual evidence from a corpora eg. Google news corpora or Wikipedia. In addition to the system predicting the veracity of a claim, the system should return documents and corresponding sentences within the documents that verify the claim. The *FEVER training Dataset* can be used as ground-truth labeled data to evaluate your system.

3. **Related Documents Retrieval:** Implement a retrieval system that retrieves related documents to a given document eg. news articles about the same event as a given news item, or academic papers on the same subject as a given paper. The *video game dataset* and it's *ground-truth labels* to can be used as ground-truth labeled data to evaluate your system. Please see *groundtruth extraction details*.

Additional bonus options may include incorporating features and advanced implementation options such as map-reduce or a simple user-interface.

1.2 Dataset

You are free to select any dataset of choice for your project. However, the datasets should be of adequate scale. Some suggested dataset sources are:

- Kaggle Q&A dataset
- Wikimedia dump
- Stackoverflow dump

1.3 Evaluation

You are required to pick or develop appropriate evaluation schemes and metrics, ideally with some ground-truth data to quantitatively evaluate your implementation. Your choice of evaluation method and its appropriateness will be factored in grading the project

1.4 Use of external libraries, ChatGPT and other assistants

You can rely on existing libraries to implement certain components eg. Lucene to retrieve all relevant documents. You are also allowed to utilize ChatGPT or other assistants for aspects of your implementation, such as the user-interface. However, all instances of the use of external resources should be referenced or documented in the report.

2 Deliverables

1. **Report:** The report in **PDF format**, to be submitted via BlackBoard. Do not submit zip-files, word documents, etc., only the submission of a single PDF file will be accepted. The report should be approximately 10 pages in length. The report should among other things include:
   (a) The problem definition; what problem is being solved?
   (b) Overview of the system built, the algorithms used, and the topics seen in class that are covered in your implementation.
   (c) The implementation details of your system.
   (d) Evaluation criteria used to evaluate the performance of your implementation. Discuss the evaluation metrics and the rationale for choosing the evaluation scheme used
   (e) Quantitative results from the evaluation metrics and discussion of results.
   (f) Limitations of your system.
2. **Implementation code of your project:** Please do not include bulky software libraries or large datasets in emails. The preferred way to share code is via a link to a publicly available GitHub repository. Include the link in your report. The files requested in the assignment description should be included in the GitHub repository. All implementation files should be placed in a ***src*** directory and all result files, if any, should be placed in a ***results*** directory.
3. **Presentation:** A short powerpoint presentation (maximum of 10 slides) to present your work. Please upload the slides to Blackboard before your presentation.
4. **Demo:** A live run of your implementation to showcase the functionality of your system.

A Note on Plagiarism

There is absolutely nothing wrong with using existing materials, you will even be commended for not reinventing the wheel, as long as you are not violating the copyright of other authors. Nevertheless, it is expected from you to clearly indicate whenever you used material that was not created by yourself. Clearly indicate in your submissions which parts constitute original work, which parts are taken from other works, and which parts were adapted from external sources. These sources have to be properly acknowledged in all your submissions. Concretely, this means at least the following guidelines are observed:

- Papers, books, webpages, blogs, etc. that were inspected while making the assignment will be referenced in a separate section "References". Citations to these materials are included in the text where appropriate.

- Text fragments exceeding one sentence that are copied from other sources are clearly marked as such. You could for instance include quoted text, definitions, etc. in italics, followed by a reference. An example of how to do this: Bela Gip (2014) defines plagiarism as *"The use of ideas, concepts, words, or structures without appropriately acknowledging the source to benefit in a setting where originality is expected"*

  References: (at the end of the document) Gipp, Bela. Citation-based plagiarism detection. Springer Vieweg, Wiesbaden, 2014. 57-88.

- When using code from other sources, indicate so in the report, and in the source code. This could for instance be done by adding a comment with a reference to the source of the function for each function that was copied from another source. It is recommended to include a separate folder "sources" in your GitHub repository with the original files from other authors that you used. Include source in the message of your commits.