Advanced Information Retrieval: Course Project

1 Introduction

Welcome to the **Advanced Information Retrieval (IR)** course project. In this project, you will work in teams of **2–3 students** to design, implement, and showcase a system that demonstrates the concepts covered throughout the course.

Project Overview

To help illustrate how each milestone might look in practice, consider the example of building a platform that **aggregates phone prices** from multiple online retailers. **However, this example is purely illustrative**—you are welcome to choose **any domain or data source** that meets the **four-milestone** structure detailed below:

1. Data Collection and Web Scraping
2. Indexing and Retrieving
3. Advanced Embedding Retrieval
4. Integration with Large Language Models (LLMs)

Additionally, you will be graded on the **quality and functionality** of your final product (i.e., the user-facing webpage or interface).

2 Project Milestones

Milestone 1: Data Collection and Web Scraping (20 points)

Objective: Gather data from multiple online sources related to your chosen domain.

Tasks:

1. Develop *web-scraping or data-collection scripts* to retrieve relevant information (e.g., item name, features, price, description).
2. Store collected data in a structured format (e.g., CSV, JSON, or a database).

3. Build a **basic webpage** (or other interface) to display the scraped data, optionally allowing users to filter or sort it.

2.0.1 deliverable:

1. Make a presentation about the results
2. show the process of collecting collecting the data
3. show the data collected on your created web app

Relevant Topics:

- *Information Retrieval Basics*: Introduction to IR and major concepts
- *Crawling and Web*: Basics of crawling, scraping, ethics, and `robots.txt` rules
- *Quality Assessment*: Ensuring data validity and handling noisy data

Milestone 2: Indexing (25 points)

Objective: Implement an indexing technique to facilitate **search queries** on the collected data.

Tasks:

1. Create a suitable index to enable fast search and retrieval.
2. Support **wildcard or approximate** searching to handle typographical errors.
3. Update the interface to include a **search bar** that uses your indexing method to retrieve relevant items.

2.1 deliverable:

1. Make a presentation about the results
2. Explain the the indexing method used
3. Explain the the retrieval method used
4. show the progress on the web app

Milestone 3: Advanced Embedding and Tree-Based Retrieval (25 points)

Objective: Enhance the system with **machine learning** techniques for item embedding and implement advanced data structures for vector-based retrieval.

Tasks:

1. Choose an **embedding model** (e.g., a pre-trained or custom ML model) to generate vector embeddings for each data entry.
2. Implement a **tree-based indexing structure** (e.g., Ball Tree, VP Tree, or similar) to facilitate nearest neighbor searches in the embedding space.
3. Update your interface to handle **semantic or similarity-based** queries, leveraging these embeddings for better retrieval accuracy.

2.2 deliverable:

1. Make a presentation about the results
2. Explain the the embedding method used
3. Explain the the retrieval method used
4. show the progress on the web app

Milestone 4: Integration with Large Language Models (LLMs) (30 points)

TBD

3 Team Structure

- **Group Size**: Each team must consist of 2–3 students.
- **Collaboration**:
  - All members should actively contribute to each milestone.
  - Collaboration tools (e.g., Git, project-management boards) are encouraged.
  - Document your contributions (who did what) to ensure transparency.

4 Final Product & Grading

Your final product is expected to be a **functioning platform** (web application or similar) demonstrating all the milestones. You will also provide a **live or recorded demo** showcasing your search interface, indexing performance, and integration of advanced techniques.

Grading Breakdown

1. Milestone 1 (Data Collection & Web Scraping): 20 points
2. Milestone 2 (Indexing with K-Grams): 25 points
3. Milestone 3 (Advanced Embedding & Tree-Based Retrieval): 25 points
4. Milestone 4 (Integration with LLMs): 30 points
5. Final Product Webpage: 20 points
    - Design & Usability: Is the interface user-friendly and visually appealing?
    - Search Performance & Speed: Does it handle queries efficiently?
    - Accuracy & Relevance: Are the results displayed correctly and sorted properly?

Total: 120 points
(Adjust due dates based on the course schedule.)

5 Deliverables

1. **Source Code**:
    - Clearly commented, uploaded to a version-control system (e.g., GitHub).
2. **Milestone Presentations**:
    - Brief written description (1–2 pages) detailing progress, challenges, and next steps for each milestone.
3. **Final Presentation/Demonstration**:
    - Show how your system handles various queries.
    - Discuss performance metrics (e.g., query time, indexing time, relevance).
4. **Final Interface**:
    - Deployed version (if feasible) or local version for demonstration.
    - Intuitive search interface and well-structured results display.

6 Getting Started

1. **Form Your Teams**:
    - Each team must have 2–3 students.
    - Submit team information (names, emails, GitHub usernames, etc.) by the first milestone presentation

2. **Choose a Domain**:
   - You can select any domain as long as you fulfill the milestone requirements.
   - Make sure your chosen project has sufficient data to explore meaningful IR challenges

3. **Seek Feedback**:
   - Discuss your approach with the instructor or TA if you have any questions.
   - Attend office hours or ask on the telegram group.

We look forward to seeing your **innovative solutions** and **functional platforms**. Good luck, and enjoy exploring the world of **Advanced IR**!