

TP-1 - Introduction to Information Retrieval

Exercise 1 - Exploring and Implementing IR Models

Instructions

1. Research four models (Boolean, Vector Space, Probabilistic, LSI) and fill a comparison table (columns: Principle, Strengths, Weaknesses, Applications).
2. Implement the **Vector Space Model** in Python using a small 3–5 document dataset.
3. Query: "information retrieval system".
4. Print document rankings and analyze why one document scores higher.

Exercise 2 - Investigating Why Information Retrieval Matters

Instructions

1. Research how IR supports **healthcare**, **e-commerce**, **academia**.
2. For each domain, note: data type, IR goal (e.g., recommendation, classification), and system used (Google Scholar, Amazon, etc.).
3. Identify one modern challenge (e.g., data scalability) and suggest a solution (e.g., semantic search).
4. Create a mind map summarizing benefits (time savings, personalization, decision support).

Exercise 3 - Modeling the Components of an IR System

Instructions

1. Research each component (document collection, indexing, query processor, ranking algorithm, UI)
2. Draw a **flowchart** showing data flow from user query → ranking → results.
3. Implement a mini simulation in Python:
 - Create 3 documents (text strings)
 - Build a basic inverted index (dictionary)
 - Input a keyword query and display matching documents

Exercise 4 - Dimensions of IR & Ad-hoc Search with Elasticsearch

Instructions

1. Research the three dimensions of IR (*Content, Applications, Tasks*) and give examples (e.g., Text + Web Search + Ad-hoc Search).
2. Compare **Elasticsearch** (IR engine) vs **NoSQL DB** (e.g., MongoDB).
3. Install Elasticsearch & Python
4. Run the following script (after starting Elasticsearch)
5. Observe ranking scores and explain how Elasticsearch implements **ad-hoc search**