

COM4014 Special Topics II

Project

Building and Analyzing a Knowledge Graph from Movie Data

In this project you need to apply concepts from the lecture series to design, construct, and analyze a knowledge graph using the [Movies dataset](#) with Neo4j and Python. The goal is to demonstrate understanding of graph modeling, querying, and advanced techniques like link prediction, node classification, and knowledge graph completion. Follow the textbook for our lecture as a guideline.

You are expected to:

1. **Design a Knowledge Graph Model**
 - Import and represent data from the Movies dataset in Neo4j
 - Identify node types (e.g., Movie, Person, Genre) and relationships
2. **Query and Explore the Graph using Cypher**
 - Perform exploratory graph analysis (e.g., degree distribution, community detection, centralities)
 - Analyze subgraphs (e.g., actor networks, genre-based groupings)
3. **Apply Advanced Graph Techniques**

Use the following techniques in your analysis:

 - Project a **monopartite** or **co-occurrence network**
 - Construct a **similarity graph** (e.g., based on shared actors or genres)
 - Perform **link prediction** to suggest collaborations or similar movies
 - Attempt **knowledge graph completion** using appropriate libraries
4. **Build a Python Script or Toolkit**
 - Use Python libraries such as scikit-learn, Neo4j, Pandas, etc.
 - Interact with your Neo4j instance through your code
 - Visualize key subgraphs, results, or learned patterns

Deliverables to submit

- A **report** summarizing:
 - Your graph model
 - Techniques applied and rationale
 - Key results, graphs, and findings
 - Any additional ideas or innovations
- A zipped **Python project folder** with clean, commented code

Notes:

1. You need to demonstrate your implementation and results with a 10-minute video demonstration. In addition, you need to submit a report (max. 5 pages) of your project results. The title page for your report must include the link to your video.
2. While showing your results, you must demonstrate all results for your findings.
3. For the parts concerning machine learning, you must use at least 3 different algorithms and compare your findings.
4. Your code should include parts that improve code readability, such as variable naming and comment lines.
5. You should develop your project in groups of no more than 2 people. All group members must be present for the demo.
6. Deadline will be announced on the E-Kampus system.

Assoc. Prof. Dr. Gazi Erkan BOSTANCI