Understanding Knowledge Graphs for Peer Learning

A **Knowledge Graph (KG)** is a structured way to represent information where **entities (users, skills, projects, etc.) are connected by relationships**. This helps in **intelligent peer matching** by analysing direct and indirect links between users.

In this project, we use a Knowledge Graph to:

✅ **Find the best peer learning partners** based on skills, projects, and coding profiles.
✅ **Identify hidden connections** between learners to create meaningful study groups.
✅ **Improve recommendations** beyond simple keyword matching.

How Does a Knowledge Graph Work?

1 **Nodes (Entities)**

- Users (people on the platform)
- Skills (Python, JavaScript, Machine Learning, etc.)
- Projects (GitHub repositories, personal projects)

2 **Edges (Relationships)**

- *User A* **has** *Skill X*
- *User B* **has** *Skill X* → A & B are connected
- *User A* **worked on** *Project Y*
- *Project Y* **requires** *Skill Z*

📌 **Example Connection:**
Alice knows Python → Bob also knows Python → Alice & Bob can be learning partners

---

Why Use a Knowledge Graph Instead of Traditional Methods?

Feature	Traditional Similarity (Cosine, Jaccard)	Knowledge Graphs
Direct Skill Matching	✅ Good	✅ Good
Finding Hidden Connections	❌ No	✅ Yes
Multi-layered Data (Projects, Interests)	❌ No	✅ Yes
Evolving & Dynamic Data	❌ Static	✅ Flexible
Better Peer Suggestions	❌ Limited	✅ Smart recommendations

How We Built This Knowledge Graph

✓ Step 1: Extract User Data
We collect user profiles from GitHub & coding platforms via **RapidAPI**.

✓ Step 2: Construct the Graph
Using **NetworkX** (Python) or **Neo4j**, we structure users, skills, and projects into a graph.

✓ Step 3: Find Peer Connections
Graph algorithms help find **the closest learning partners** based on **direct & indirect relationships**.

✓ Step 4: Recommend Study Groups
Using **clustering techniques**, we suggest groups where users can collaborate efficiently.

Conclusion

A **Knowledge Graph** makes learning **more interactive, connected, and intelligent**. It goes beyond simple skill-matching and **creates a dynamic ecosystem** where learners can **find the best peers based on skills, projects, and experience**.

🎯 **This approach enhances peer-to-peer learning and helps users grow efficiently!**