

Knowledge Graphs

A Beginner's Workshop

Association of Data Science

Workshop Agenda

1 What are Knowledge Graphs?

Concepts and applications

2 Building Your First Graph

Hands-on with NetworkX

3 Querying and Analysis

Extracting insights

4 RDF & Semantic Web

Industry standards

5 Real-World Uses

Applications

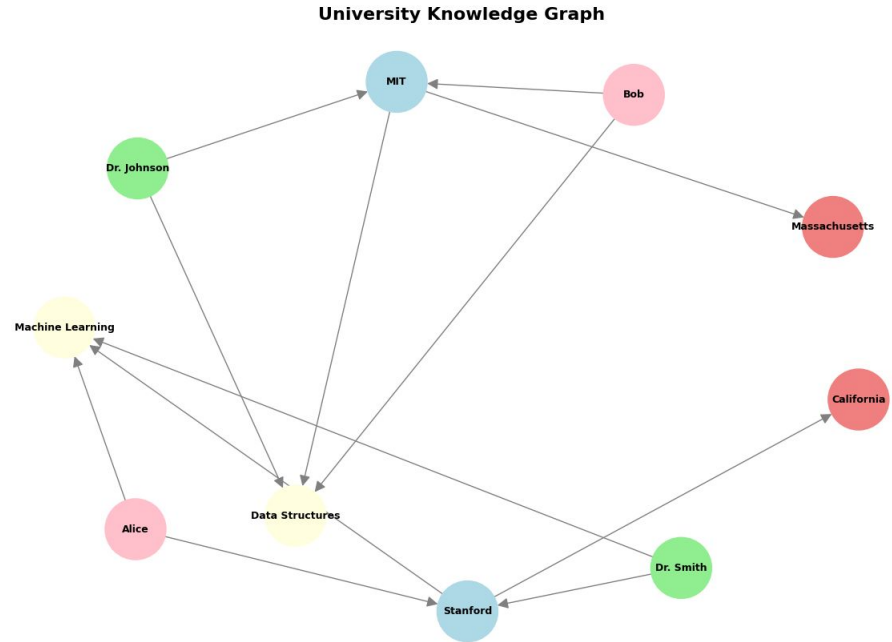
What is a Knowledge Graph?

Definition

A network of entities (nodes) and relationships (edges) representing information in a structured, interconnected way.

Core Components

- Nodes: Entities (people, places, concepts)
- Edges: Relationships between entities
- Properties: Attributes and metadata



Knowledge Graphs in the Wild

Google Search

Powers results with entity understanding and direct answers

Netflix

Recommendation engine based on content relationships

Finance

Fraud detection through transaction pattern analysis

Facebook

Social Graph connects billions of users and relationships

Healthcare

Links diseases, symptoms, drugs for drug discovery

AI & ML

RAG systems and embeddings for LLMs

Building Your First Graph

Using NetworkX

```
# Create graph
kg = nx.DiGraph()

# Add nodes
kg.add_node("Alice", type="Student")

# Add relationships
kg.add_edge("Alice", "Stanford")
```

Key Concepts

- Nodes represent entities
- Edges capture relationships

Visualization Tools

Pyvis

Interactive HTML visualizations with drag-and-drop

NetworkX + Matplotlib

Static visualizations for publications

Gephi

Standalone tool for large-scale analysis

Querying Knowledge Graphs

What is Querying?

Querying a knowledge graph means asking questions and extracting specific information from the network of connected entities. Unlike traditional databases where you query tables, with knowledge graphs you traverse relationships to discover connections and patterns.

Entity Lookup

Find all nodes of a specific type (e.g., all students, all courses)
Filter entities by their properties or attributes
Example: "Find all students enrolled in Machine Learning"

Find Entities

```
students = [n for n in kg.nodes()]
```

Relationship Traversal

Navigate from one entity to connected entities
Follow edges to discover relationships
Example: "What courses is Alice taking?" → Follow edges from Alice node

Path Finding

```
nx.shortest_path(kg, "Alice", "CA")
```

Discover hidden patterns in your data

RDF

RDF

W3C standard using triples:
Subject-Predicate-Object

Core Philosophy: Everything can be represented as a statement with three parts:

- **Subject:** The thing being described
- **Predicate:** The property or relationship
- **Object:** The value or another resource

Triple Example

Alice → age → 21

Why RDF?

- Industry standard
- Interoperability

SPARQL

Query language for RDF (like SQL for graphs)

Tools & Technologies

Python

NetworkX

RDFLib

Graph

Neo4j

Apache Jena

ML & Advanced

PyTorch Geometric

Node2Vec

All tools are open source and free

Next Steps & Resources

Learning

Stanford CS224W

Neo4j Graph Academy

NetworkX Documentation

Project Ideas

Movie recommendation system

Research citation network

Course prerequisite mapper

Thank You!

Ready to build your own knowledge graphs?

Key Takeaways

- Knowledge graphs connect entities through relationships
- NetworkX makes graphs easy to build and query
 - Real-world applications are everywhere
- Open source tools are accessible to everyone

Questions?

Association of Data

Science