

1 Thư viện thuật toán đồ thị

1.1 Neo4j Graph Data Science (GDS)

Neo4j GDS hỗ trợ nhiều thuật toán:

- Shortest Path: Dijkstra, BFS, A*, Yen
- Centrality: PageRank, Betweenness, Closeness
- Community Detection: Louvain, Label Propagation
- Similarity: Jaccard, Cosine
- Embedding: GraphSAGE, Node2Vec, FastRP

1.2 Python NetworkX

NetworkX là thư viện đồ thị trong Python, dùng để triển khai các thuật toán từ dữ liệu trích từ Neo4j.

2 Chạy thuật toán Shortest Path trong Neo4j GDS

Dưới đây là truy vấn chạy thuật toán Dijkstra giữa hai cầu thủ:

```
MATCH (src:Player {id: 'player_david_raya'})  
MATCH (dst:Player {id: 'player_bukayo_saka'})  
CALL gds.shortestPath.dijkstra.stream('epl_graph', {  
    sourceNode: id(src),  
    targetNode: id(dst)  
})  
YIELD totalCost, nodeIds  
RETURN  
    totalCost,  
    [nodeId IN nodeIds | gds.util.asNode(nodeId).id] AS path;
```

Kết quả hiển thị trong Hình 1.

The screenshot shows the neo4j browser interface running on localhost:7474/browser/. It displays two main sections:

- Top Section:** Shows the command `neo4j$ MATCH (src:Player {id: 'player_david_raya'}) MATCH (dst:Player {id: 'player_bukayo_saka'}) CALL gds.shortestPath.dijkstra.stream('epl_graph')`. Below it, a table shows the result of the shortest path query:

totalCost	path
2.0	["player_david_raya", "club_arsenal", "player_bukayo_saka"]

Text and Warn buttons are visible on the left.
- Bottom Section:** Shows the command `neo4j$ CALL gds.graph.project('epl_graph', ['Player', 'Club', 'Season'], { PLAYED_FOR: {orientation: 'UNDIRECTED'}, PARTICIPATED_IN: {orientation: 'UNDIRECTED'} })`. Below it, a table shows the results of the graph projection:

nodeProjection	relationshipProjection	graphName	nodeCount	relationshipCount	projectMillis
{ "Player": { "label": "Player", "properties": {} }, }	{ "PARTICIPATED_IN": { "aggregation": "DEFAULT", "orientation": "UNDIRECTED", "indexInverse": false, "properties": {} }, }	"epl_graph"	781	1898	2331

Text, Code, and a gear icon are visible on the left.

The system tray at the bottom shows a weather icon (15°C), a power icon (Có máy), and the date/time (11/18/2025 3:18 PM).

Hình 1: Kết quả thuật toán Dijkstra trong GDS

3 Chạy Shortest Path bằng Python NetworkX

Mã Python sử dụng để tính đường đi ngắn nhất:

```
from neo4j import GraphDatabase
import networkx as nx

driver = GraphDatabase.driver(
    "neo4j://localhost:7687",
    auth=("neo4j", "test1234")
)

query = """
MATCH (p:Player)-[:PLAYED_FOR]->(c:Club)
RETURN p.id AS player, c.id AS club
"""

G = nx.Graph()

with driver.session() as session:
    for record in session.run(query):
        G.add_edge(record["player"], record["club"])

src = "player_david_raya"
dst = "player_bukayo_saka"

path = nx.shortest_path(G, src, dst)
print("Shortest path:", path)
```

Kết quả minh họa trong Hình 2.

The screenshot shows a terminal window with several tabs open. The current tab contains Python code for connecting to a Neo4j database and calculating the shortest path between two players using NetworkX. The code is as follows:

```
from neo4j import GraphDatabase
import networkx as nx

driver = GraphDatabase.driver("neo4j://localhost:7687", auth=("neo4j", "test1234"))

query = """
MATCH (p:Player)-[:PLAYED_FOR]->(c:Club)
RETURN p.id AS player, c.id AS club
"""

G = nx.Graph()

with driver.session() as session:
    for record in session.run(query):
        G.add_edge(record["player"], record["club"])

src = "player_david_raya"
dst = "player_bukayo_saka"

path = nx.shortest_path(G, src, dst)
print("Shortest path:", path)
```

The terminal output shows the execution of the code and the resulting shortest path:

```
Shortest path: ['player_david_raya', 'club_arsenal', 'player_bukayo_saka']
```

Hình 2: Kết quả chạy thuật toán shortest path bằng Python NetworkX