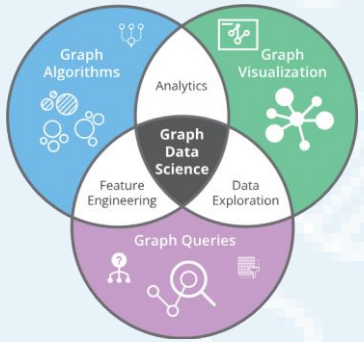

Habib University
shaping futures

CS343 Graph Data Science

Spring 2024

Paths and Shortest Paths

Muhammad Qasim Pasta

qasim.pasta@sse.habib.edu.pk

Slides are intended to be filled during the lectures. Certain details are intentionally omitted for in-class discussions. These slides are not meant to be used as reading material.

Graph Data Science

Uses multi-disciplinary **workflows** that may include statistics, queries, algorithms, and machine learning.

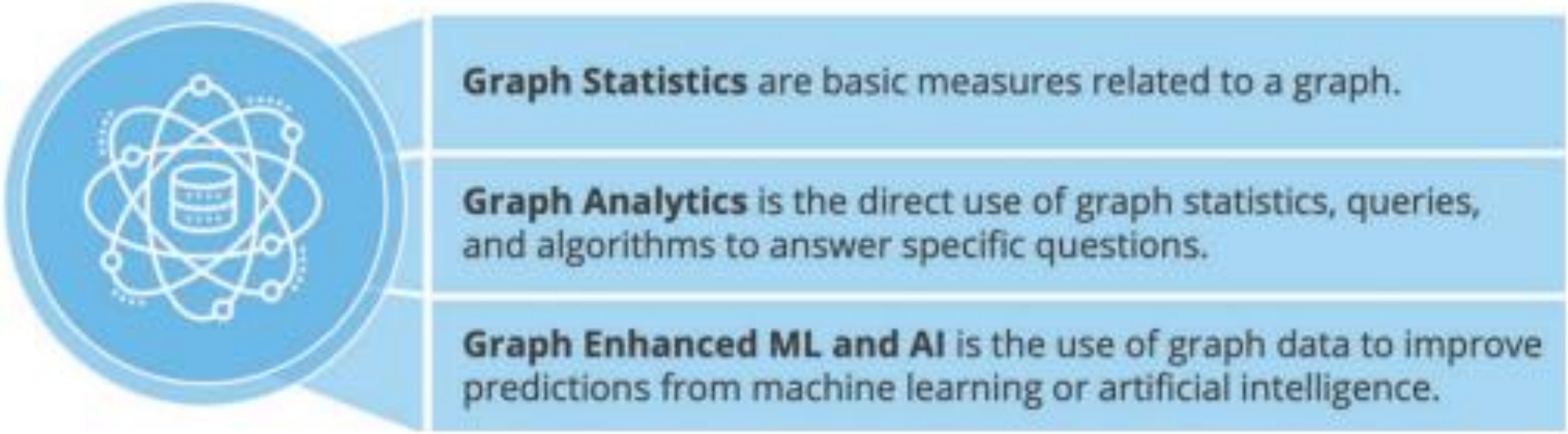

Image: Graph Data Science Using Neo4j

Graph Analytics

- Analysing and extracting insights from graph
- Try to infer meaning based on the network structure: various algorithms, techniques, and methods to understand the structure, properties, and patterns within graphs.

Path Analytics

- analysing paths or routes between nodes
  - finding shortest path
  - common paths
  - detecting patterns in the sequences of nodes
- Transportation Planning:
  - shortest routes between locations to optimize logistics and transportation networks.
- Network Security:
  - potential attack paths in computer networks to identify vulnerabilities.
- Recommendation Systems:
  - user navigation paths to improve recommendations in e-commerce or content platforms.

Connectivity Analytics

- Analysing the connectivity patterns within a graph
  - identifying connected components,
  - measuring the overall connectivity
  - robustness of the graph.
- Network Robustness Analysis:
  - Resilience of a network to failures or attacks by analysing its connectivity properties
- Transportation Planning:
  - identify disconnected regions or islands within the network.

Community Analytics

- Identifying communities or groups of based on topological properties demonstrating similar behaviours
- Partition graph into groups based on certain criteria
- Marketing:
  - market segments or customer groups with similar purchasing behaviour for targeted advertising campaigns.
- Fraud Detection:
  - groups of colluding entities in financial networks by identifying tightly-knit communities.
- Collaboration Networks:
  - Analysing co-authorship networks in academia to identify research communities and trends.

Centrality Analytics

- Quantify the importance or influence of nodes within a graph
- Some centrality metrics: degree, betweenness, and eigenvector
- Influencer Marketing:
  - influential individuals in social networks for targeted marketing campaigns.
- Urban Planning:
  - critical infrastructure nodes such as transportation hubs or power distribution centres.
- Epidemiology:
  - individuals with high betweenness centrality in disease transmission networks to prioritize vaccination efforts.

Graph Traversing in Cypher

Graph Traversing in Cypher

Movie	

```
MATCH (p:Person)-[:ACTED_IN]->(m:Movie)
WHERE p.name = 'Eminem'
RETURN m.title as Movie
```

Graph Traversing in Cypher

Path in Cypher

- We can assign a variable to path

```
MATCH q = (m:MyNode{Name:'A’}) -[*]- (n:MyNode{Name:"E"})
RETURN COUNT(q)
```

  - This will return the number of paths between A and E

```
MATCH q = (m:MyNode{Name:'A’}) -[*]- (n:MyNode{Name:"E"})
RETURN length(q)
```

  - This will return the length of each path between A and E

Path in Cypher

- We can limit the length of paths we are interested in

```
MATCH q = (m:MyNode{Name:'A’}) -[*2]- (n:MyNode{Name:"E"})
RETURN COUNT(q)
```

  - This will return paths of size 2 between nodes A and E

```
MATCH q = (m:MyNode{Name:'A’}) -[*2..5]- (n:MyNode{Name:"E"})
RETURN COUNT(q)
```

  - This will return paths of having length between 2 and 5 starting from A and ending at E

Path in Cypher: Shortest Paths

- We have ShortestPath() and AllShortestPath() functions – we can provide maximum but not minimum

```
MATCH q = ShortestPath(m:MyNode{Name:'A’})
          -[*]- (n:MyNode{Name:"E"}))
RETURN COUNT(q)
```

  - Returns the shortest path between node A and E

```
MATCH q = AllShortestPaths(m:MyNode{Name:'A’})
          -[*]- (n:MyNode{Name:"E"}))
RETURN COUNT(q)
```

  - Returns all shortest paths between node A and E (must be of same size)

Practice

- Returns the shortest path between the Eminem node and the Charlton Heston node
- Returns the shortest path between the Eminem node and the Charlton Heston node considering only ACTED_IN relationship
- Retrieve all Person nodes that are exactly two hops away from Eminem using the ACTED_IN relationship.
- Retrieve all Person nodes that are up to four hops away from Eminem using the ACTED_IN relationship.
- Return a list of actors that are up to 6 hops away from Tom Hanks