Welcome To

Met Connect

By Helin Yilmaz, Jordan Andersen, Oviya Adhan, Tiffany Liu

DATASCI 205, Section 1, Group 4

Business Case Scenario

We joined the digital innovation team at

The Met as data scientists!

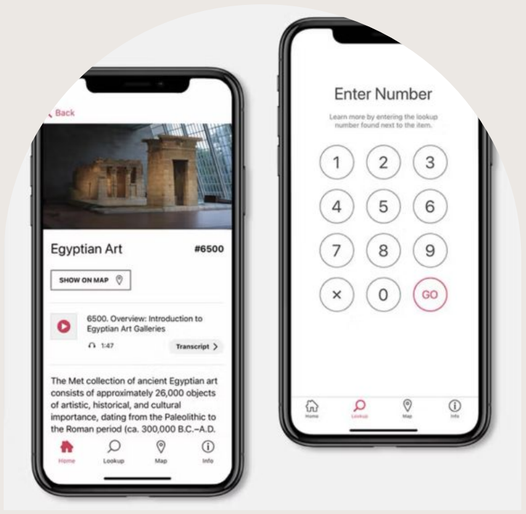
- 470,000+ artworks
- In-person + online

Business Vision

Design a data-driven system that…

Powers recommendation engine

Showcases artwork connections



Improves web / mobile content delivery

Generates ideas for art exhibits / research

NoSQL Databases We Use

Neo4j

(Graph Data Science Algorithms)

MongoDB

Redis

Graph Design

Nodes

- Artist
- Artwork (Paintings only)
- Department
- Time Period

Edges

Artwork–Artist

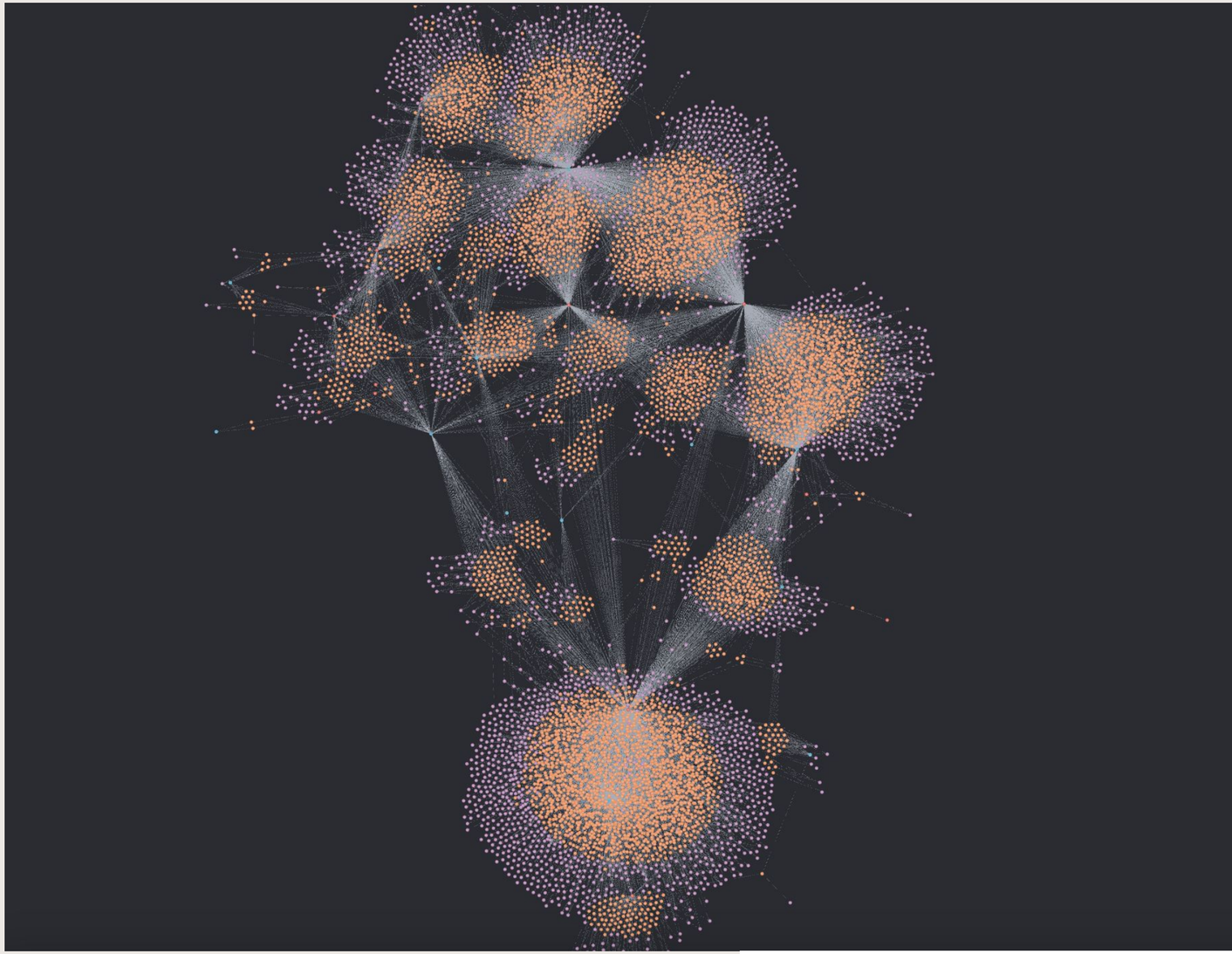
(:Artwork)-[:CREATED_BY]->(:Artist)

Artwork–Department

(:Artwork)-[:BELONGS_TO]->(:Department)

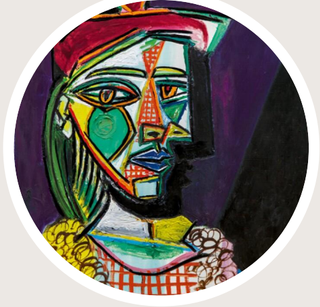
Artwork–Time Period

(:Artwork)-[:ASSOCIATED_WITH]->(:TimePeriod)



```
------------------------
 Nodes: 8607
 Relationships: 17844
------------------------
```

Neo4j: Pagerank



Goals:

- Identify the most influential artists
- Identify the centuries with the most influential artists

Result:

- Famous artists such as John Singer Sargent, Claude Monet, and Pablo Picasso had the highest Pagerank scores
- Artists from the 18th century had the highest average Pagerank scores

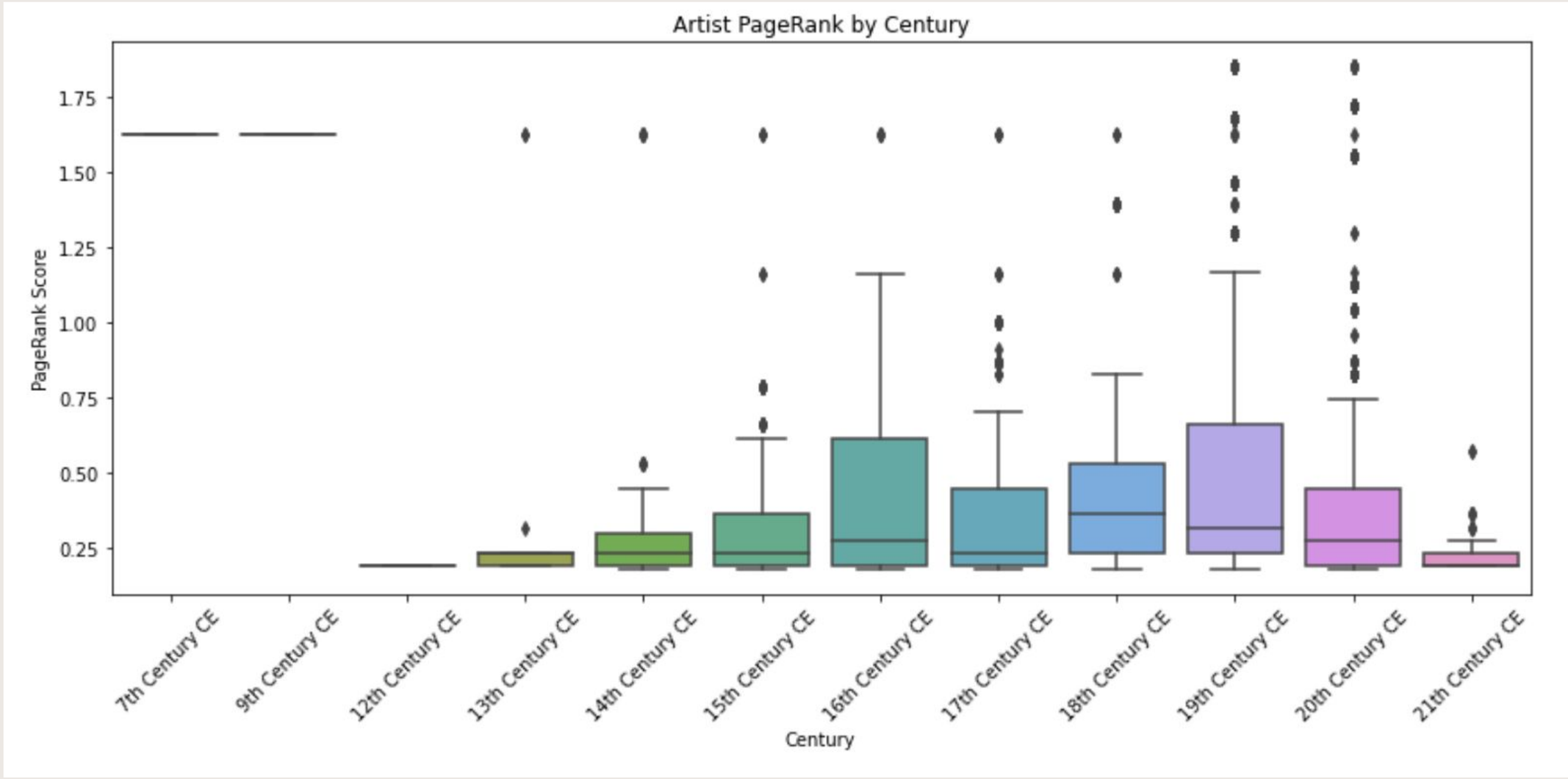
Business Case Applications:

- Identifies current artist rankings to determine future art acquisitions

Why Not Relational Database:

- Computationally intensive for a relational database

Neo4j: PageRank



Neo4j: Closeness Centrality

Goal: Identify the most centralized art departments

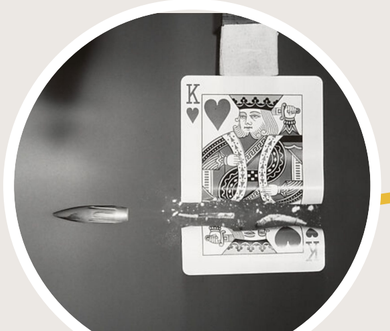
Result: European Paintings are the most centralized with a closeness centrality score of 0.3446

Business Case Applications:

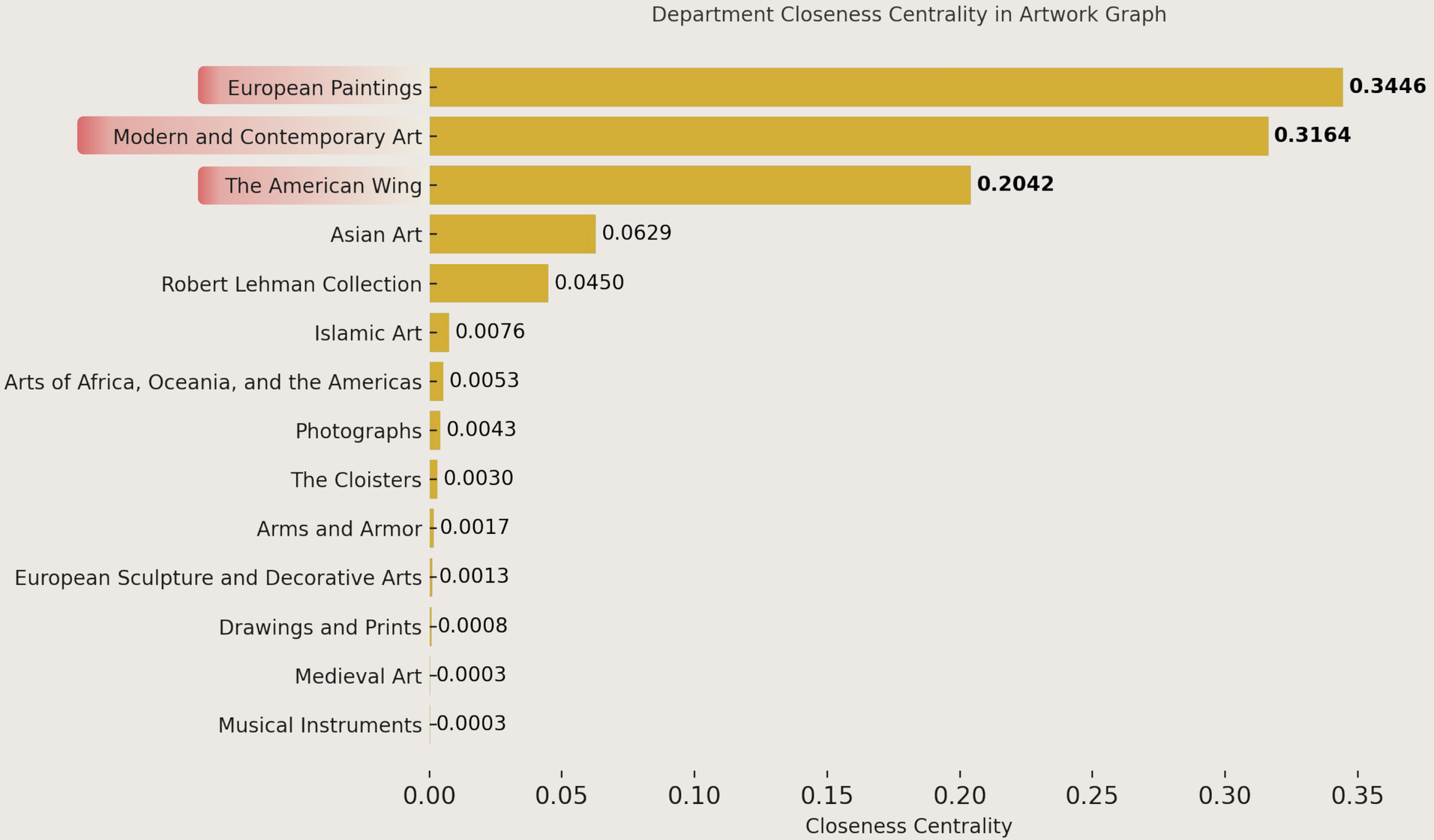
- Identifies possible relationships for art history research
- Prompts collaboration on cross-departmental exhibits

Why Not Relational Database:

- Shortest Path calculation is computationally expensive & memory-intensive
- Less scalable for complex datasets



Neo4j: Closeness Centrality



Neo4j: Louvain Modularity

Goal: Detect communities connected by departments/time periods

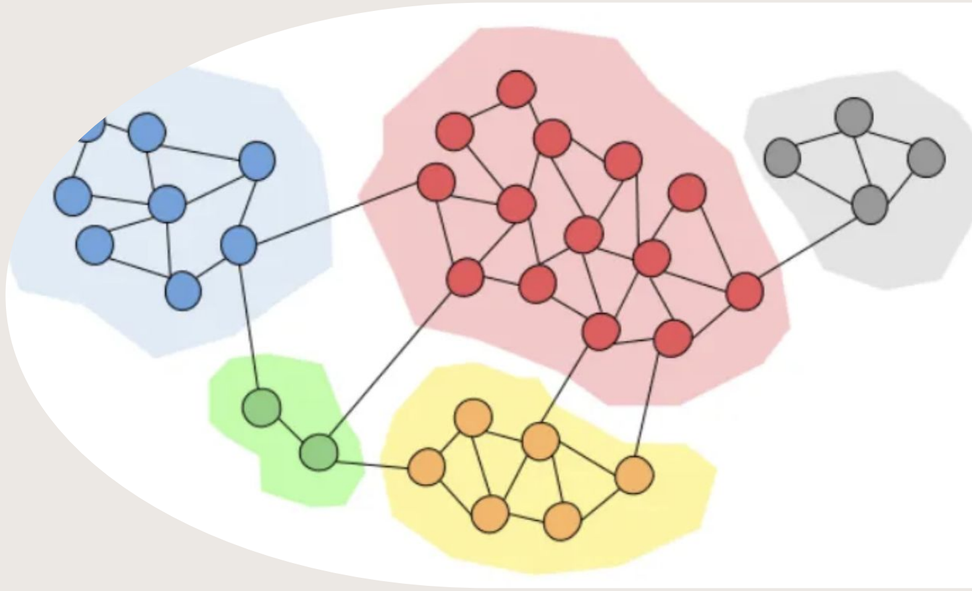
Result: 9 communities, 0.61 modularity score

Business Case Applications:

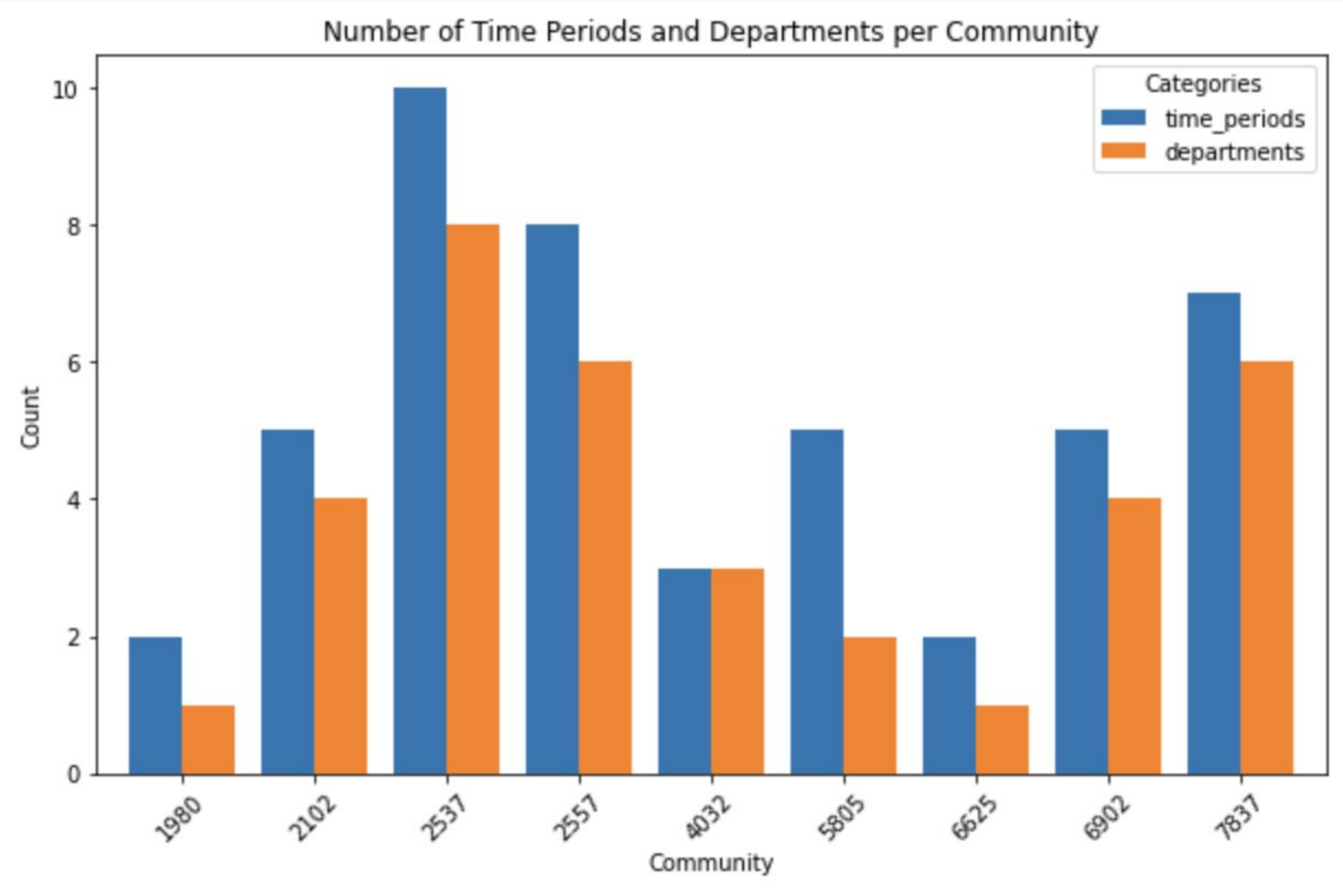
- Reveals relationships ideal for storytelling in tours
- Helps recommend related artworks/artists
- Supports exhibits on movements within a community

Why Not Relational Database:

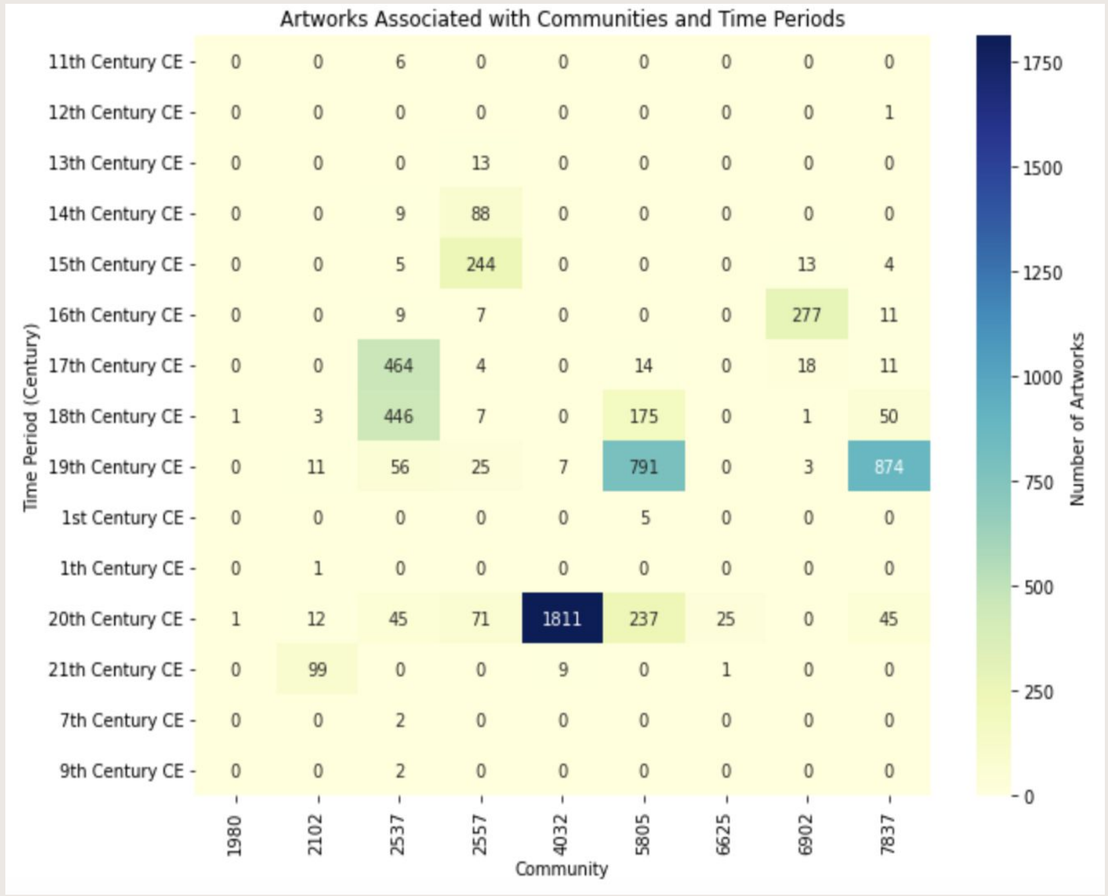
- Struggle with deeply interconnected data
- Require expensive joins

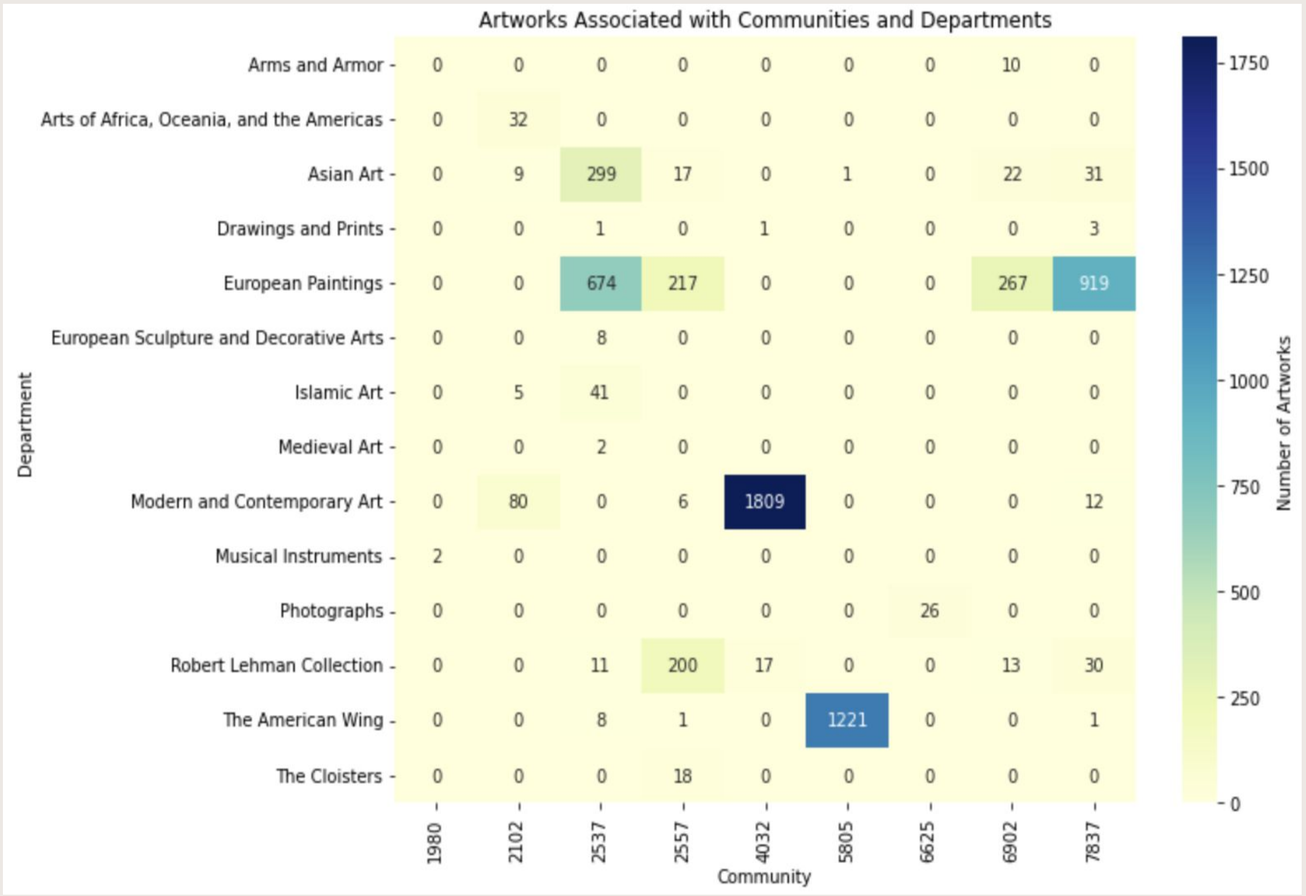


Neo4j: Louvain Modularity



Neo4j: Louvain Modularity





MongoDB

Goal

- Generate exhibition ideas based on visitor interests

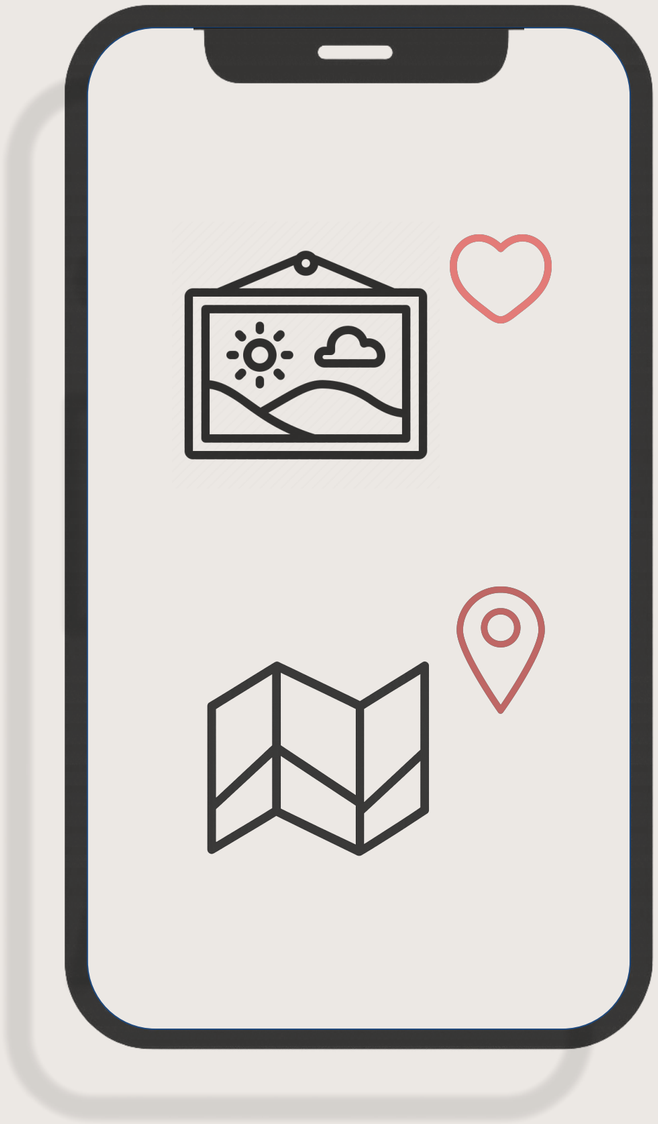
Method

- Visitors can participate in a survey at the end of their visit to rate their favorite departments, artworks, and artists that they saw that day.
- Create points of view (POVs) for artists, artwork, departments, and time periods.
- Store and query this data with Mongo DB for future analysis when showcasing new exhibits.

Why not a relational database

- Requires flexibility to analyze our data from multiple POVs

Redis



Goal: Create a mobile, real-time recommendation system for visitors

Method:

- Visitors favorite artworks on mobile app
- Visitors' location tracked by mobile app
- App uses favorites and/or live location in museum (key) to provide recommendations of artwork currently on view in nearby galleries (value)

Why Not Relational Database:

- Requires immediate access to *subsets* of the database
- Cannot provide *dynamic* updates as needed

References

The Met Dataset

Our GitHub Repo

Q&A

Any Questions?