Graph Query Portal

AMIT DAYAL
DAVID BROCK
CS4624: MULTIMEDIA, HYPertext, AND INFORMATION ACCESS
INSTRUCTOR: EDWARD A. FOX
VIRGINIA TECH, BLACKSBURG, VA, 24061
MAY 1, 2018
Outline

- Problem Description
- Challenges
- System Architecture
- Data Flow
- Scenario Description
  - Live Demo
- Extra Features
- Acknowledgments
- References
- Questions
Problem Description:

Background

- Social Interactome project
  - Set of social network-based clinical trials to test interventions to improve network effect on addiction recovery.

Problem

- How do we make the data analysis process easier?
Challenges

- Data preparation is hard, time consuming. (MySQL schema is messy).
- Analysis across heterogeneous sources and info is difficult.
- Analysis process is not currently stream-lined and/or reproducible.
Aims

- Identify key, repeated tasks executed by SI researchers.
- Model data and operations to support these tasks.
System Architecture Cont.

Graph Query Portal Data
Storage: Neo4j

Middleware: Node.js
Data Flow

HTTP Response → Neo4j Results

Processing

Neo4j

HTTP Request → Query
Live Demo - Scenarios

As a researcher:

1. I want to see a single participant’s data
2. I want to see all participant data
3. I want to see all the nodes in this graph
4. I want to see the amount of engagements between two people
5. I want to see the number of engagements between people and compare it with another network’s engagements
Extra Features

- Added a Mongo Database instance for logging API and Neo4j interactions
- Created a Front-end website to display information from the API in a browser
  - Can interact with data
  - Support for all read APIs
Acknowledgments

Clients: Prashant Chandrasekar, Dr. Fox

NIH Grant 1R01DA039456-01: The Social Interactome of Recovery: Social Media as Therapy Development
Questions
References

- "node-v0.x-archive on GitHub". Retrieved 2 August 2014.
- Neubauer, Peter [@peterneubauer] (17 Feb 2010). "@sarkkine #Neo4j was developed as part of a CMS SaaS 2000-2007, became released OSS 2007 when Neo Technology spun out"(Tweet) – via Twitter.
- neo4j (February 9, 2018). "Release 3.3.3" – via Github.
Appendix:

See Following Slides for Live Demo Information
Scenario 1: I want to see a single participant’s data

```
```
Scenario 2: I want to see all the participants’ data

```json
Scenario 2: I want to see all the participants’ data
```

![HTTP request](https://via.placeholder.com/150)

```
Method: GET
Request URL: http://localhost:5007/e1r1/participant
```

### Parameters

<table>
<thead>
<tr>
<th>Headers</th>
<th>Authorization</th>
<th>Variables</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Toggle source mode</td>
<td>Insert headers set</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Header name** | **Header value**
--- | ---

**ADD HEADER**

- **Headers are valid**
- **565.64 ms**
- **200 OK**

```json
Array(167)
  -0: {
    "nodeID": 280,
    "label": "Array1",
    0: "PROFILE",
    "properties": {
      "country": "",
      "language": "",
      "sex": "",
      "education": "",
      "pub_keywords": ""
    }
  }
```

---

**DETAILS**

- **Headers size:** bytes

---

![HTTP response](https://via.placeholder.com/150)
Scenario 3: I want to see the graph in its entirety
Scenario 4: As a researcher, I would like to see the amount of engagements between two people
Scenario 5: I want to see the number of engagements between people and compare it with another network’s engagements