Challenges:

The main challenge encountered was to upload a very large dataset into neo4j database. Neo4j Client takes forever to load the entire dataset on our machine configuration. It does require some latest hardware like SSD’s and high capacity of RAM in order to load the entire dataset and generate nodes for the users and jokes and creates relationship between them. We also tried some performance tuning that is mentioned below but it marginally improved the performance of the entire process. For demo purpose, we are limiting our dataset to 50,000 records and we provide joke recommendations to a particular user from it.

Performance Tuning:

We have tuned certain parameters in order to maximize the performance. They are listed as follows:

1. Page Cache Sizing:

* Page Cache sizing provided caching the Neo4j data. This provided avoiding costly access to the disk. Thus making sure that the performance is not degraded.
* This can be implemented by setting the “dbms.pagecache.memory” property in neo4j.conf file.

1. Increase heap size:

* The heap size is also increased from its default value to sustain multiple operations simultaneously.
* This does provide a significant performance boost for Neo4j
* The can be implemented by setting dbms.memory.heap.initial\_size and dbms.memory.heap.max\_size in neo4j.conf.

Advantages of using Neo4j:

* Neo4j uses Cypher queries for graph traversal and manipulation that is easier to learn, understand and write.
* Provides different graph representation like Visual Graph representation, row representation and text representation.
* There are many different ways we can perform operations on graph using the Neo4j library. It can be used in different programming languages like Java and Python; It can also be used in command shells like Gremlin shell and neo4j-shell. It can also be used on their own Neo4j Client.
* Visual representation of the graph provides better understanding of the relationships between the nodes.
* It is easy and faster to traverse through the graph.

Disadvantages of using Neo4j:

* Loading large dataset requires having a machine with better hardware.
* Does not support Sharding.

Outputs:

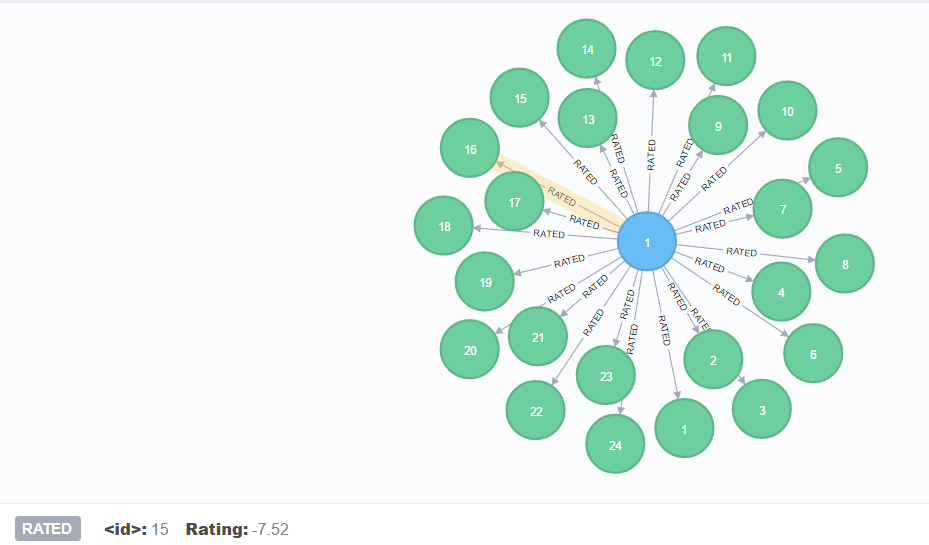


Fig 1.1: Graph for user with Id 1 consisting of ratings relationship with some jokes.

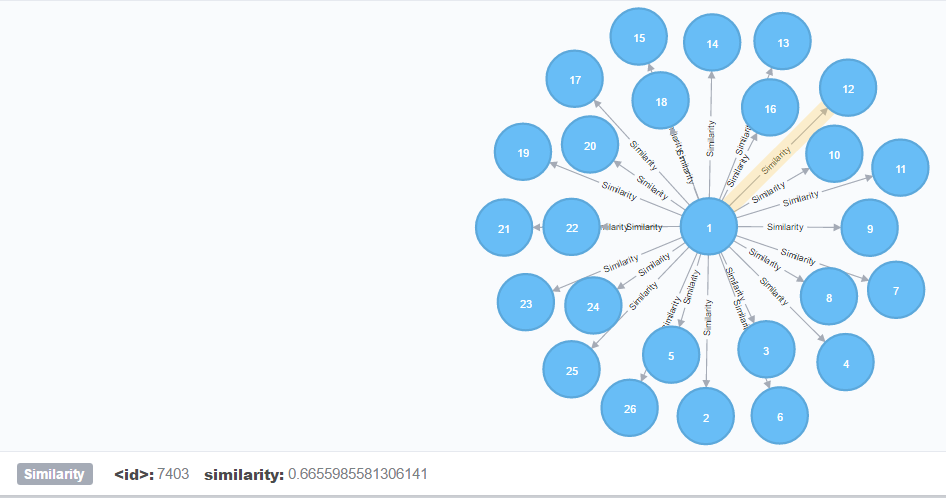


Fig 1.2: Similarity relationships for the current user having ID 1 with other similar users

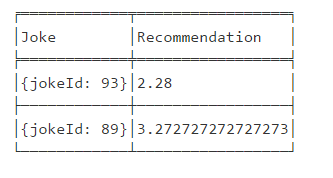


Fig 1.3: Recommended Jokes with their IDs and their average ratings.

References:

1. Neo4j Performance Tuning - <https://neo4j.com/developer/guide-performance-tuning/>
2. Beer Recommender System - <http://mikelam.azurewebsites.net/beer-recommendations-with-user-based-collaborative-filtering/>