

Tutorial Running Code

The below document will serve as a README file, this will a walkthrough tutorial on what is needed and how to run the application that has been accomplished during the project.

Requirements:

- The credentials to gain access to the Server where the data is being held.
 - During the process of the project, I was provided the EngineB server credentials to investigate and undertake the project. However, I have removed the credentials from the code because of Intellectual Property reasons.
- Neo4j enterprise local server on your local machine together with its credentials.
 - You will need to download the neo4j enterprise server edition (has a 1-month free trial). This localhost is used to project the algorithmic results isolated in a neo4j environment.
- The correct dependencies being installed to allow access to specific libraries.
 - The dependencies are already included in the **pom.xml** file, the *projects code folder neo4j-algo* already has all these dependencies applied for you. However, you will still need to install them in the correct project directory. This can be done manually, by installing the jar files. An easier way is to open the project file in IDLE (Eclipse, intellij) and use the IDLE help to install the dependencies when building the project.
 - Dependencies needed is **org.neo4j.driver** with version **4.2.1**, this dependency will give us access to neo4j java libraries and will allow us to interact with neo4j database.
- The correct Java version to be compatible with the 4.2.1 neo4j java driver. You should have a java version smaller than 15. Ideally **Java version 14.0.1** should be used during the process of running the code.

Steps to Run Code:

1. Make sure all the Requirements are fulfilled before attending to run the code.
2. You will need to initialize a connection with your local neo4j enterprise server from terminal or cmd.
 - a. Open terminal or cmd
 - b. Navigate yourself in the neo4j-enterprise file directory (You have downloaded)

```
~ % cd Desktop/University/THESIS/CODE/Dummie-Server neo4j-enterprise-4.2.7 %
```

- c. Initialize and start a server connection with your local server using command **./bin/neo4j console**

```

louca5z@192 ~ % cd Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7
louca5z@192 neo4j-enterprise-4.2.7 % ./bin/neo4j console
WARNING! You are using an unsupported Java runtime.
* Please use Oracle(R) Java(TM) 11, OpenJDK(TM) 11 to run Neo4j.
* Please see https://neo4j.com/docs/ for Neo4j installation instructions.
Directories in use:
home: /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7
config: /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7/conf
logs: /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7/logs
plugins: /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7/plugins
import: /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7/import
data: /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7/data
certificates: /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7/certificates
run: /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7/run
Starting Neo4j.
2021-07-28 10:52:30.748+0000 INFO Starting...
2021-07-28 10:52:33.558+0000 INFO ===== Neo4j 4.2.7 =====
2021-07-28 10:52:36.096+0000 INFO Sending metrics to CSV file at /Users/louca5z/Desktop/University/THESIS/CODE/Dummie-Server/neo4j-enterprise-4.2.7/metrics
2021-07-28 10:52:36.131+0000 INFO Bolt enabled on localhost:7687.
2021-07-28 10:52:37.636+0000 INFO Remote interface available at http://localhost:7474
2021-07-28 10:52:37.637+0000 INFO Started.

```

3. We now have the connection open
4. Modify the servers you will be using during the application run.
 - a. Modify server credentials in Class **Connector** on line **150** , fill in the credentials of the server where you will be obtaining the data from.

```

149 public static void main(String... args) throws Exception {
150     try (Connector graphDB = new Connector(url: [redacted], user: [redacted], password: [redacted])) {
151         // Get the number of nodes in the graph.
152         List<Record> number_nodes = graphDB.executeSimpleQuery(sub_query_relation);
153         //printSingleNodes(number_nodes);
154
155         // Apply query to all nodes and edges
156         List<Record> full_graph = graphDB.executeSimpleQuery(main_query_relation);
157         //printAllGraphResults(full_graph);
158     }
159 }

```

Fill in the red boxes with the correct credentials

- b. Modify the credentials in class **Connector** on lines [184, 199, 228, 257] , fill the credentials with your local server where you would like to project the results of the pipelines.
5. Once all these steps have been accomplished, we can now run the application by running class **Connector**.

```

/Library/Java/JavaVirtualMachines/jdk1.8.0_191.jdk/Contents/Home/bin/java ...
Jul 28, 2021 2:03:00 PM org.neo4j.driver.internal.logging.JULLogger info
INFO: Direct driver instance 1595953398 created for server address [redacted]
There are two pipelines implemented, choose the following :
[1] Pipeline -> Weakly Connected Components + Degree Centrality.
[2] Pipeline -> Label Propagation + Degree Centrality.
[3] Pipeline -> Label Propagation + Degree Centrality considering biased nodes first in changing labels.
[4] Pipeline -> Label Propagation + Degree Centrality with ONLY the biased nodes from ML
Type the appropriate number from the options [1, 2, 3, 4].
Option 3 & 4 work for query that includes nodes of type [Datapoint or Ratio]

```

6. You now need to type the option you would like to run from [1, 2, 3, 4] as shown in the screenshot. The procedure will start applying the algorithm and in the end project the results on your local server.

```

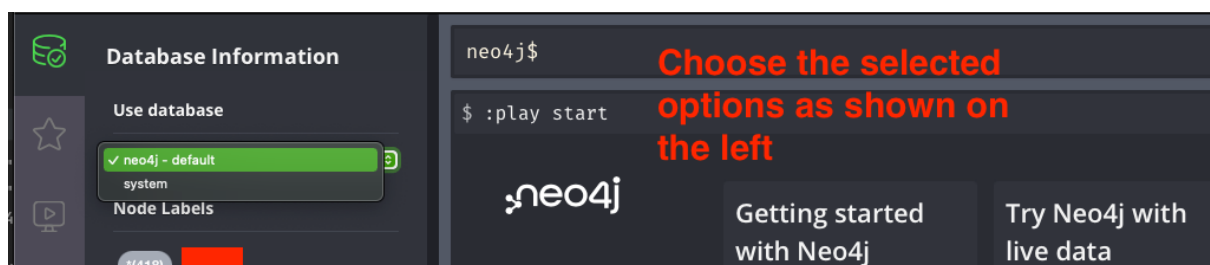
Node: 5128 Node: 4801 Node: 4802 Node: 5059 Node: 5060 Node: 4741 Node: 4681 Node: 4621 Node: 4622 Node: 4951 Node: 4891 Node: 4892 Node: 4831 Node: 5089 Node: 4771 Node: 4711 Node: 4712 Node: 4651
[1 Popular node][Areas][["1", "Fixed Assets"] has degree power 24
[2 Popular node][Datapoint, TotalAssets][["Total AssetsMarvell_2016", "Assets", "1", 4648650000, "Fixed Assets", "Marvell_2016", "Total Assets"] has degree power 1
[3 Popular node][Datapoint, OtherNonCurrentAssets][["Other Non Current AssetsMarvell_2015_Q3", "Fixed Assets", "1", 100304000, "Marvell_2015_Q3", "Other Non Current Assets"] has degree power 1
[4 Popular node][Datapoint, TotalAssets][["Total AssetsMarvell_2015_Q3", "Assets", "1", 5506524000, "Fixed Assets", "Marvell_2015_Q3", "Total Assets"] has degree power 1
[5 Popular node][Datapoint, OtherNonCurrentAssets][["Other Non Current AssetsMarvell_2014", "Fixed Assets", "1", 128839000, "Marvell_2014", "Other Non Current Assets"] has degree power 1
[7] Community Label is: 414
Size of community is :16
Node: 4897 Node: 4867 Node: 5125 Node: 4837 Node: 4887 Node: 5095 Node: 4777 Node: 5065 Node: 4747 Node: 4717 Node: 4687 Node: 4657 Node: 4627 Node: 5050 Node: 4957 Node: 4927
[1 Popular node][Areas][["1", "Ownership"] has degree power 10
[2 Popular node][Datapoint, TotalShareholderEquity][["Total Shareholder EquityMarvell_2016_Q2", "Ownership", "1", 4159080000, "Marvell_2016_Q2", "Total Shareholder Equity"] has degree power 1
[3 Popular node][Datapoint, TotalShareholderEquity][["Total Shareholder EquityMarvell_2016_Q1", "Ownership", "1", 4099810000, "Marvell_2016_Q1", "Total Shareholder Equity"] has degree power 1
[4 Popular node][Datapoint, TotalShareholderEquity][["Total Shareholder EquityMarvell_2016", "Ownership", "1", 4027651000, "Marvell_2016", "Total Shareholder Equity"] has degree power 1
[5 Popular node][Datapoint, TotalShareholderEquity][["Total Shareholder EquityMarvell_2015_Q4", "Ownership", "1", 4140123000, "Marvell_2015_Q4", "Total Shareholder Equity"] has degree power 1
Jul 28, 2021 2:06:06 PM org.neo4j.driver.internal.logging.JULLogger info
INFO: Direct driver instance 174573182 created for server address
DELETED DATA
Jul 28, 2021 2:06:33 PM org.neo4j.driver.internal.logging.JULLogger info
INFO: Closing driver instance 174573182
Jul 28, 2021 2:06:33 PM org.neo4j.driver.internal.logging.JULLogger info
INFO: Closing connection pool towards
Jul 28, 2021 2:06:33 PM org.neo4j.driver.internal.logging.JULLogger info
INFO: Closing driver instance 1595953398
Jul 28, 2021 2:06:33 PM org.neo4j.driver.internal.logging.JULLogger info
INFO: Closing connection pool towards

```

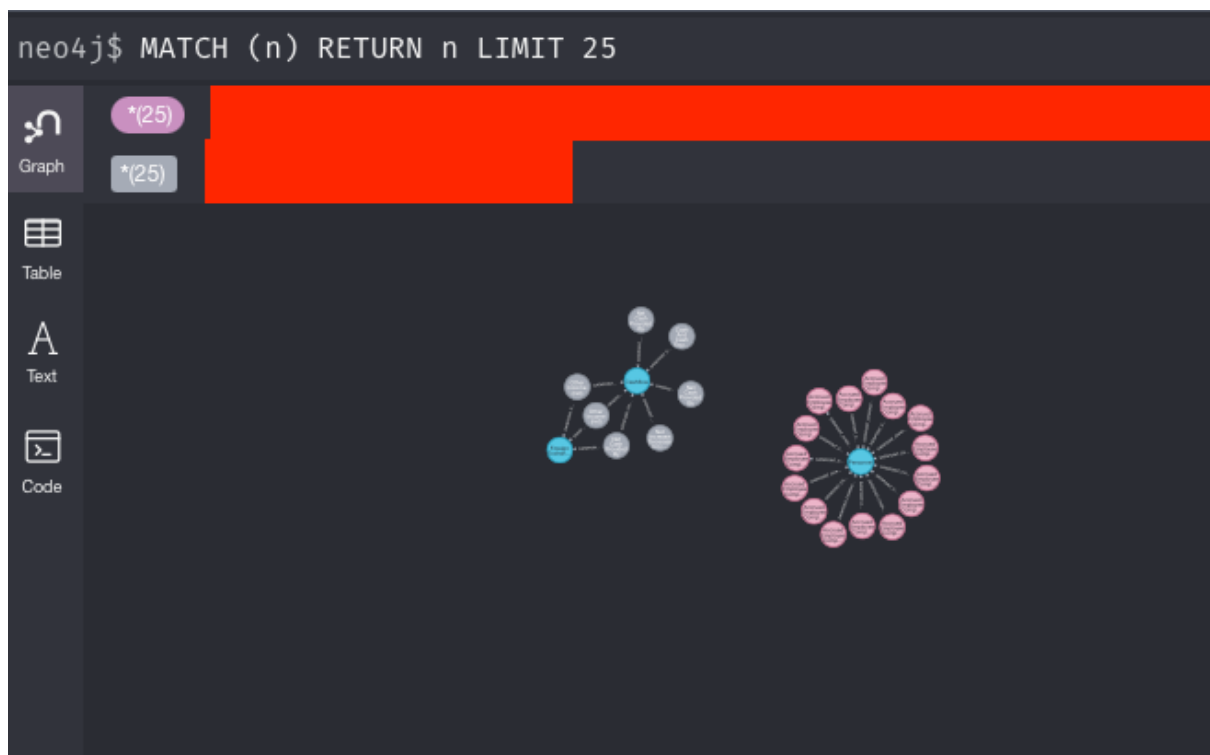
Text results of communities and degree powers

Run has been accomplished

- Now navigate in your browser to URL : localhost:7474 to view your local neo4j server and preview the results in visualisation rather than text.



- Apply a neo4j query **MATCH (n) RETURN n LIMIT 25** to preview the results.



This is the end of the tutorial and procedure walkthrough. With the same steps you can apply each algorithm at a time on a different relation to investigate. If you would like to investigate a specific use case of interest you will need to modify the filtering queries in

Class **Connector** on lines **[68 – 79]** to obtain the appropriate data, you would like to apply the pipelines on.