|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Machine information | | | | |
| Type | Cpu | Memory | VolSize | OS |
| m5.2xlarge | 8 | 32GB | 200GB | ubuntu18 |

RESULT:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Loading time | convert | WCC | PageRank |
| TigerGraph | 1119s |  | 23558ms | 267882ms |
| Neo4j | 1163s | 1670s | 50016ms | 643968ms |

We will test more algorithms for the benchmark e.g. bfs, cosine similarity...

Instruction

1. **Generate 30GB LDBC dataset.**

**1.1 Configuration**

cd ./ldbc\_snb\_datagen

Initialize the params.ini file as needed. For example, to generate the basic CSV files, issue:

cp params-csv-basic.ini params.ini

**1.2 Install Java, python2.7 and maven**

If you haven’t install Java please run the following script:

wget --no-check-certificate --no-cookies --header "Cookie: oraclelicense=accept-securebackup-cookie" <http://download.oracle.com/otn-pub/java/jdk/8u131-b11/d54c1d3a095b4ff2b6607d096fa80163/jdk-8u131-linux-x64.tar.gz>

Move jdk1.8.131 to /usr/local

tar -zxvf jdk-8u131-linux-x64.tar.gz

mv jdk1.8.0\_131 jdk1.8

mv jdk1.8 /usr/local

Install python2.7 and maven

sudo apt update

sudo apt install python-minimal

sudo apt install maven

**1.3 Set environment values**

To configure the amount of memory available, set the HADOOP\_CLIENT\_OPTS environment variable. To grab Hadoop, extract it, and set the environment values to sensible defaults, run the following script:

cp params-csv-basic.ini params.ini

wget http://archive.apache.org/dist/hadoop/core/hadoop-3.2.1/hadoop-3.2.1.tar.gz

tar xf hadoop-3.2.1.tar.gz

export HADOOP\_CLIENT\_OPTS="-Xmx2G"

# set this to the Hadoop 3.2.1 directory

export HADOOP\_HOME=`pwd`/hadoop-3.2.1

# set this to the repository's directory

export LDBC\_SNB\_DATAGEN\_HOME=`pwd`

export JAVA\_HOME=/usr/local/jdk1.8

**1.4 Change the parameter.**

vi params.ini

Change the first line to

ldbc.snb.datagen.generator.scaleFactor:snb.interactive.30

Then run the following command to start generation.

./run.sh

The data will be stored at /ldbc\_snb\_datagen/social\_network

Then move the data. Here change ubuntu to your user name

mkdir /home/ubuntu/test\_data

mv ./social\_network/static /home/ubuntu/test\_data

mv ./social\_network/dynamoc /home/ubuntu/test\_data

1. **Create Schema and load data for tigergraph**

Then we will go back to the home folder. First we need to edit the filename in gsql-shell.

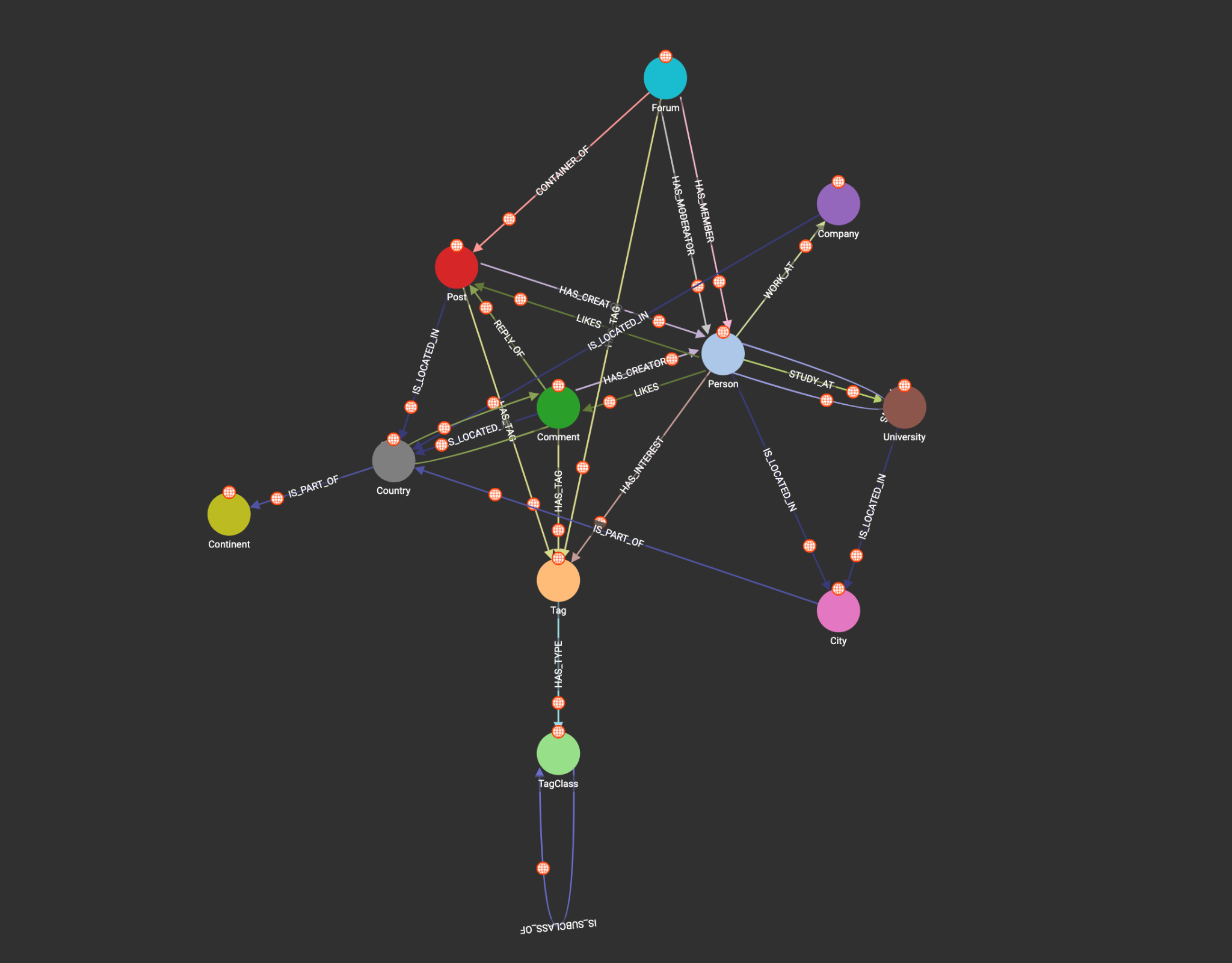
Edite line 42-74. Change the path to your actual location of the LDBC data. If you have move the data to test\_data folder, you can only change ubuntu to your user name. The line you need to edit shows in the below picture.



Then run gsql schema.gsql.

You can also enid the create and load query at schema.docx. Then you can copy the query to gsql-shell and run it.(To open gsql-shell, type gsql in terminal)

After the loading job, you can open GraphStudio to check if the schema looks like the picture below.



1. **Test algorithm for TigerGraph.**

Install algorithm using conn\_comp.gsql and pageRank.gsql(in the folder)

gsql conn\_comp.gsql

gsql pageRank.gsql

Then run the following commend:

time gsql run.gsql

The file run.gsql is also in the main folder.

1. **Create and load data for neo4j**

Go to the cypher folder.

cd ./ldbc\_snb\_implementations/cypher/

Run the following command to initialize Neo4j.

./get-neo4j.sh

./environment-variables-neo4j.sh

./configure-neo4j.sh

export NEO4J\_HOME=’pwd’/neo4j-server

export NEO4J\_DB\_DIR=’pws’/neo4j-server/data/databases/graph.db

**Preprocessing and loading.**

Set the following environment variables appropriately, you need to change the directory of NEO4J\_DATA\_DIR(change ubuntu to your username).

export NEO4J\_HOME=’pwd’/../neo4j-server

export NEO4J\_DB\_DIR=$NEO4J\_HOME/data/databases/graph.db

# set according to your data source

export NEO4J\_DATA\_DIR=/home/ubuntu/test\_data

export POSTFIX=\_0\_0.csv

Then run the following command to convert and load data.

./convert-csvs.sh

./delete-neo4j-database.sh

./import-to-neo4j.sh

1. **Test algorithm for Neo4j**

First we need to copy the algo-library to Neo4j plugins. Go back to the main folder.Copy the graph-algorithms-algo-3.5.4.0.jar to neo4j-server/plugins

mv graph-algorithms-algo-3.5.4.0.jar ./ldbc\_snb\_implementations/neo4j-server/plugins/graph-algorithms-algo-3.5.4.0.jar

Then go to the ./ldbc\_snb\_implementations/neo4j-server/conf. Add the the following line to ./neo4j.conf

dbms.security.procedures.unrestricted=algo.\*

Go back to ./neo4j-server/bin run the following command:

./neo4j start

./cypher-shell

Your username:**neo4j**

Password:**admin**

Run following command to test WCC and pageRank:

CALL algo.pageRank.stream('Comment','REPLY\_OF')

YIELD nodeId,score;

CALL algo.unionFind.stream('Comment'','REPLY\_OF')

YIELD nodeId,setId;