CTA200H Final Project Writeup

Anatoly Zavyalov

May 14, 2021

1 Abstract

2 Installation & Setup

The following steps outline how to install JanusGraph along with the Apache Cassandra storage backend[1] and the Elasticsearch indexing backend[2], as well as setting up gremlin-python, which is used for querying the JanusGraph backend from a Python interface. The operating system used is **Ubuntu 20.04.2 LTS**.

2.1 Installing Java

JanusGraph is built on top of Apache Tinkerpop [3], which, in turn, is built on top of Java and hence requires Java SE 8. The implementation of Java that we will install is OpenJDK 1.8. First, we refresh the list of available packages:

```
$ sudo apt update
```

Next, we install OpenJDK 1.8:

```
$ sudo apt install openjdk-8-jre
```

To verify that the correct version has been installed, we run `java -version`. We should see something similar to `openjdk version "1.8.0_292"`.

2.2 Setting the `$JAVA_HOME` environment variable

Next, we must set the `$JAVA_HOME` environment variable. First, we head to `/usr/lib/jvm/` and locate the intallation of the JDK. It should look similar to `/usr/lib/jvm/java-11-openjdk-amd64`. Next, we set the `$JAVA_HOME` environment variable to point to the installation of the JDK:

```
$ export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
```

We doublecheck that this is successful with `echo $JAVA_HOME`.

[1] https://cassandra.apache.org/

[2] https://www.elastic.co/elasticsearch/

[3] https://tinkerpop.apache.org/

2.3 Setting up JanusGraph

From the JanusGraph Releases[4], we download the .zip of the “full” installation of JanusGraph (the file name should resemble `janusgraph-full-X.X.X.zip`, where `X.X.X` is the version number), and extract the contents. This “full” installation includes JanusGraph, as well as pre-configured Apache Cassandra and Elasticsearch.

From here, we start the JanusGraph server by running

```
$ bin/janusgraph.sh start
```

We can then open the Gremlin console by running

```
$ bin/gremlin.sh
```

Next, we may create a remote connection to the JanusGraph server:

```
gremlin> :remote connect tinkerpop.server conf/remote.yaml
```

From here, we can send commands to the JanusGraph server by preceding them with `:>`. We can avoid this by running

```
gremlin> :remote console
```

which will enable sending all queries directly to the JanusGraph server and avoid the need of `:>`.

3 Timestamp System

4 Scaling

5 Further Discussion

[4] https://github.com/JanusGraph/janusgraph/releases