Lab 1

Big Data, Spring 2018

The goal of this lab is to get you set up with the machines we will be using this semester and to execute a parallel program using Hadoop-Streaming.

Before starting this lab, you will need your HPC account username and password. You should have received an email with these.

Part 1: HPC@NYU

First, we will get set up with NYU's high-performance computing (HPC) infrastructure, documented at http://wikis.nyu.edu/display/NYUHPC.

Windows users: If you don't have "Windows Subsystem for Linux", you will need to download the PuTTY utilities: http://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html.

Step 1: HPC@NYU tutorial

Please follow the official HPC@NYU Tutorial #1, available at
http://wikis.nyu.edu/pages/viewpage.action?pageId=53859101

Note that there are many other tutorials and instructions for accessing and using the HPC clusters available at https://wikis.nyu.edu/display/NYUHPC.

Part 2: Running a Hadoop job on Dumbo

Next, we will experiment with the Hadoop cluster (Dumbo). In particular, we will explore the Hadoop-Streaming utility, documented at
http://hadoop.apache.org/docs/current/hadoop-streaming/HadoopStreaming.html
Hadoop-Streaming exposes a subset of Hadoop's functionality to other programming languages (Hadoop uses Java). We will use Hadoop-Streaming to execute MapReduce programs written in Python.

Information about and instructions for using the dumbo cluster can be found here:
https://wikis.nyu.edu/display/NYUHPC/Clusters+-+Dumbo.

Step 1: Log in to Dumbo

From **within the NYU Network** (on-campus or VPN)

1. Connect using SSH:
   Mac OS, Linux, Windows 10 with Linux Subsystem
   1. Open a terminal
   2. Enter the following command: `ssh netid@dumbo.hpc.nyu.edu` replacing `netid` by your NetID

   Windows (older versions)
   1. Install PuTTY: https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html
   2. Run PuTTY.exe.
   3. In the “Host Name” field, enter: `netid@dumbo.hpc.nyu.edu` replacing `netid` by your NetID.
   4. Click “Open” at the bottom.
2. Enter your password when prompted.
3. Once you see a prompt, check that you are running Bash (one of several "shells"). The prompt should say something like “-bash-4.1$”. If not, type “echo $0” and it should say “-bash”. If it says something else, like “tsch”, type “bash” to switch to bash. (You are welcome to use another shell, but you’ll need to modify the following instructions, so we don’t recommend it unless you really know what you’re doing.)

From **outside the NYU Network**, it requires a 2-step process:

Mac OS, Linux, Windows 10 with Linux Subsystem

1. Open a terminal
2. Enter the following command: `ssh netid@gw.hpc.nyu.edu` replacing `netid` by your NetID, and enter your password when prompted
3. Enter the following command: `ssh dumbo.hpc.nyu.edu` and enter your password again when prompted.

Windows (older versions)

1. Run PuTTY.exe (assuming you've already downloaded it).
2. In the “Host Name” field, enter: `netid@gw.hpc.nyu.edu` replacing `netid` by your NetID.
3. Click “Open” at the bottom and enter your password when prompted
4. At the prompt, enter the following command: `ssh dumbo.hpc.nyu.edu` and enter your password when prompted.

You are now logged in to Dumbo. Type `pwd` to see what directory you are currently in (it should be ‘/home/`netid`’.

Step 2: Create aliases for Hadoop-Streaming

The default commands to run Hadoop-Streaming are lengthy and it saves time to create aliases.

1. On Dumbo, edit your .bashrc file, using your editor of choice. For example:
   using vi: `vi ~/.bashrc`
   using emacs: `emacs ~/.bashrc`
   using nano: `nano ~/.bashrc`
   (If you haven't used any of these editors before, nano is the easiest starting place.)

2. Add the following lines to .bashrc:
```
HADOOP_EXE='/usr/bin/hadoop'
HADOOP_LIBPATH='/opt/cloudera/parcels/CDH/lib'
HADOOP_STREAMING='hadoop-mapreduce/hadoop-streaming.jar'

alias hfs="$HADOOP_EXE fs"
alias hjs="$HADOOP_EXE jar $HADOOP_LIBPATH/$HADOOP_STREAMING"
```

3. Now type
   ```
   source ~/.bashrc
   ```
   to introduce these aliases into your current environment. Note that Bash 'sources' .bashrc automatically at login, so you won't have to type this command again.

Step 3: Running a Hadoop-Streaming Job

In this part, we will run an example MapReduce program on Dumbo using Hadoop-Streaming.

The example program we will run is a wordcount program. We give a large data file as input ("book.txt"), and the program counts the number of times each word occurs in the text.

1. First, we will create a folder in our home directory to store the files and output. Type
   ```
   mkdir -p ~/example
   ```

2. Now navigate to the folder you just created:
   ```
   cd ~/example
   ```

3. Copy the example into this directory:
   ```
   cp -r /share/apps/examples/hadoop-streaming/* .
   ```
   (note the trailing dot -- this expands to the name of the current directory)

4. Look at the input files. To see the contents that were copied to the folder, type `ls`.

You should see the input data “book.txt”, a README file, and a directory called “src”. This directory is where the code is. You can look at the code for the map function by typing `cat src/mapper.py` and `cat src/reducer.py`

5. Copy the input text file to HDFS: we need to move the book.txt file to the HDFS file system. To do this, type:

```
hfs -put book.txt
```

6. **Make sure to clear out output from previous runs** (otherwise your job will fail as HDFS will not overwrite the existing output file):

```
hfs -rm -r example.out
```

7. Run the Hadoop-Streaming job:

```
hjs \
-D mapreduce.job.reduces=2 \
-file ~/example/src \
-mapper  src/mapper.sh \
-reducer src/reducer.sh \
-input  /user/netid/book.txt \
-output /user/netid/example.out
```

Note that the backslashes "\" must be the last character before the newline --- no intervening whitespace --- for Bash to understand to continue parsing on the next line.

8. After the job has finished, move the output from HDFS to your home directory. If you type

```
hfs -get example.out
```

you will now have a directory called ‘example.out’ in the current directory. To see the contents of this directory type `ls example.out` . Notice that it contains two files (“part-00000” and “part-00001”) - this is because 2 reduce tasks were used. To view these files, you can type, for example, `cat example.out/part-00000`

If you want the output in one file, Hadoop can merge it for you. Type

```
hfs -getmerge example.out examplemerged.txt
```

You should then have a text file called “examplemerged.txt” which contains all the output. To view it type `cat examplemerged.txt`

9. Now we will move “examplemerged.txt” from our home directory on Dumbo to our own local machine. (These steps assume you are connected via the NYU network)

   Mac OS, Linux, Windows 10 with Linux Subsystem:
   Enter the following command into a terminal:
   ```
   scp netid@dumbo.hpc.nyu.edu:~netid/example/examplemerged.txt .
   ```

   Windows (older versions)
   1. Install PSCP (comes with PuTTY, or can download it separately from PuTTY website)
   2. Run cmd.exe.
   3. Navigate to the folder where you saved pscp.exe
   4. Enter the command

   ```
   pscp netid@dumbo.hpc.nyu.edu:~netid/example/examplemerged.txt .
   ```

   The file examplemerged.txt is now on your local computer.

   Deliverable: (Due Monday, 2/5/18 at 6pm) Submit your examplemerged.txt file to NYU Classes -> Assignments -> Lab 1

Part 3: Amazon Web Services (AWS)

We will lastly set up an AWS account. Feel free to use an existing account if you have one.

1. Sign up for an AWS account: http://aws.amazon.com
2. Sign up for AWS Educate (you will need this for $$ credits): https://aws.amazon.com/education/awseducate/apply/
3. Generate a public/private key-pair. Follow the instructions at: http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html#having-ec2-create-your-key-pair
   For PuTTY (Windows) users, you will need to convert your .pem file to a .ppk file. Follow instructions at: http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html#putty-private-key