CS246: Mining Massive Datasets **Winter 2016**

Hadoop Tutorial

Due 11:59pm January 12, 2016

General Instructions

The purpose of this tutorial is (1) to get you started with Hadoop and (2) to get you acquainted with the code and homework submission system. Completing the tutorial is optional but by handing in the results in time students will earn 5 points. This tutorial is to be completed individually.

Here you will learn how to write, compile, debug and execute a simple Hadoop program. First part of the assignment serves as a tutorial and the second part asks you to write your own Hadoop program.

Section 1 describes the virtual machine environment. Instead of the virtual machine, you are welcome to setup your own pseudo-distributed or fully distributed cluster if you prefer. Any version of Hadoop that is at least 1.0 will suffice. (For an easy way to set up a cluster, try Cloudera Manager: `http://archive.cloudera.com/cm5/installer/latest/cloudera-manager-installer.bin`.) If you choose to setup your own cluster, you are responsible for making sure the cluster is working properly. The TAs will be unable to help you debug configuration issues in your own cluster.

Section 2 explains how to use the Eclipse environment in the virtual machine, including how to create a project, how to run jobs, and how to debug jobs. Section 2.5 gives an end-to-end example of creating a project, adding code, building, running, and debugging it.

Section 3 is the actual homework assignment. There are no deliverable for sections 1 and 2. In section 3, you are asked to write and submit your own MapReduce job

This assignment requires you to upload the code and hand-in the output for Section 3.

All students should submit the output via GradeScope and upload the code via snap.

GradeScope: To register for GradeScope,

- Create an account on GradeScope if you don't have one already.
- Join CS246 course using Entry Code 92B7E9

Upload the code: Put all the code for a single question into a single file and upload it at `http://snap.stanford.edu/submit/`.

Questions

1 Setting up a virtual machine

- Download and install *VirtualBox* on your machine: http://virtualbox.org/wiki/Downloads
- Download the *Cloudera Quickstart VM* at https://downloads.cloudera.com/demo_vm/virtualbox/cloudera-quickstart-vm-5.5.0-0-virtualbox.zip.
- Uncompress the VM archive. It is compressed with 7-zip. If needed you can download a tool to uncompress the archive at http://www.7-zip.org/.
- Start *VirtualBox* and click *Import Appliance* in the *File* dropdown menu. Click the folder icon beside the location field. Browse to the uncompressed archive folder, select the .ovf file, and click the *Open* button. Click the *Continue* button. Click the *Import* button.
- Your virtual machine should now appear in the left column. Select it and click on *Start* to launch it.
- To verify that the VM is running and you can access it, open a browser to the URL: http://localhost:8088. You should see the resource manager UI. The VM uses port forwarding for the common Hadoop ports, so when the VM is running, those ports on localhost will redirect to the VM.
- *Optional*: Open the Virtual Box preferences ($File \rightarrow Preferences \rightarrow Network$) and select the *Adapter 2* tab. Click the *Enable Network Adapter* checkbox. Select *Host-only Adapter*. If the list of networks is empty, add a new network. Click *OK*. If you do this step, you will be able to connect to the running virtual machine via SSH from the host OS at 192.168.56.101. The username and password are 'cloudera'.

The virtual machine includes the following software

- CentOS 6.4
- JDK 7 (1.7.0_67)
- Hadoop 2.5.0
- Eclipse 4.2.6 (Juno)

The virtual machine runs best with 4096MB of RAM, but has been tested to function with 1024MB. Note that at 1024MB, while it did technically function, it was very slow to start up.

2 Running Hadoop jobs

Generally Hadoop can be run in three modes.

1. **Standalone (or local) mode:** There are no daemons used in this mode. Hadoop uses the local file system as an substitute for HDFS file system. The jobs will run as if there is 1 mapper and 1 reducer.
2. **Pseudo-distributed mode:** All the daemons run on a single machine and this setting mimics the behavior of a cluster. All the daemons run on your machine locally using the HDFS protocol. There can be multiple mappers and reducers.
3. **Fully-distributed mode:** This is how Hadoop runs on a real cluster.

In this homework we will show you how to run Hadoop jobs in Standalone mode (very useful for developing and debugging) and also in Pseudo-distributed mode (to mimic the behavior of a cluster environment).

2.1 Creating a Hadoop project in Eclipse

(There is a plugin for Eclipse that makes it simple to create a new Hadoop project and execute Hadoop jobs, but the plugin is only well maintained for Hadoop 1.0.4, which is a rather old version of Hadoop. There is a project at `https://github.com/winghc/hadoop2x-eclipse-plugin` that is working to update the plugin for Hadoop 2.0. You can try it out if you like, but your milage may vary.)

To create a project:

1. Open Eclipse. If you just launched the VM, you may have to close the Firefox window to find the Eclipse icon on the desktop.
2. Right-click on the *training* node in the Package Explorer and select *Copy*. See Figure 1.

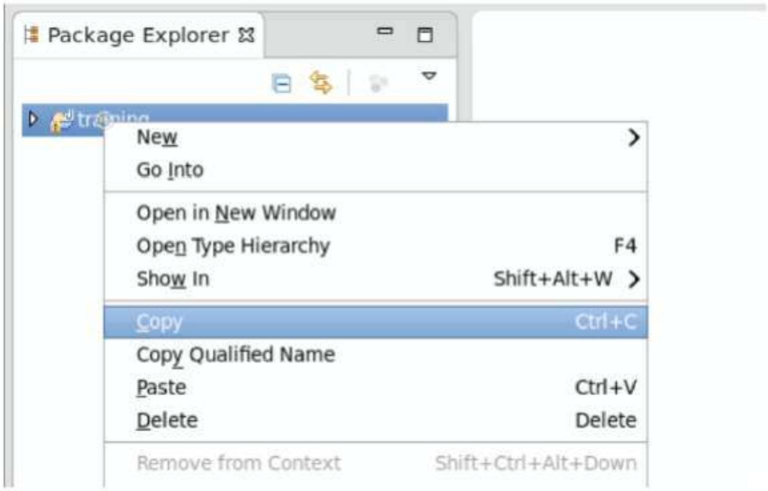


Figure 1: Create a Hadoop Project.

3. Right-click on the *training* node in the Package Explorer and select *Paste* . See Figure 2.

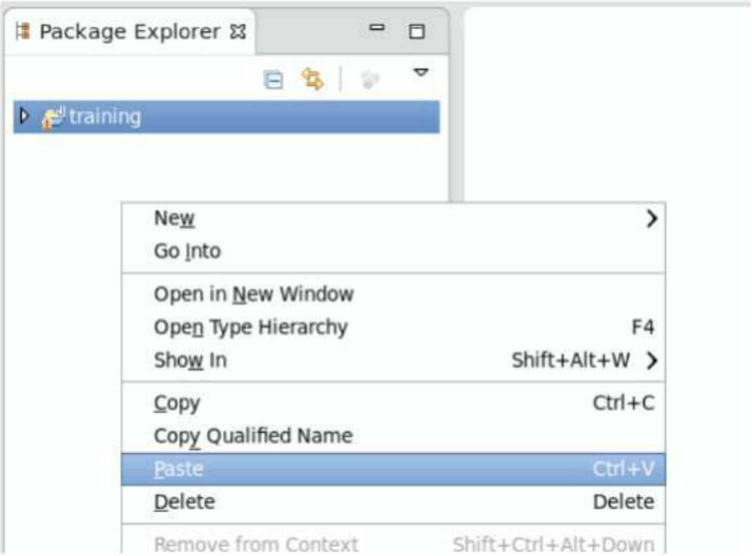
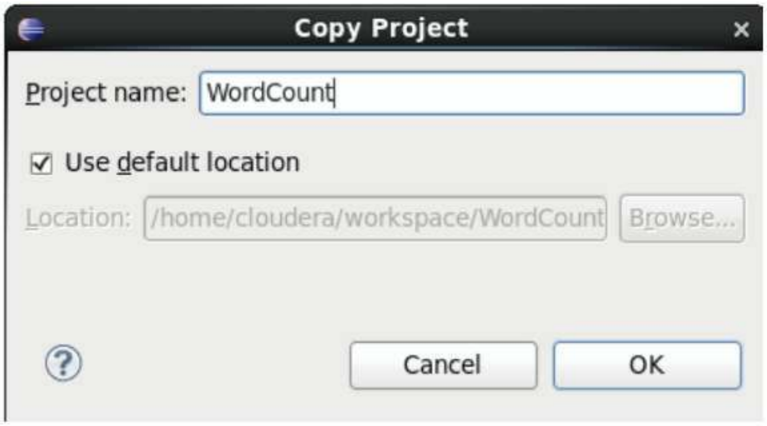


Figure 2: Create a Hadoop Project.

4. In the pop-up dialog, enter the new project name in the *Project Name* field and click *OK*. See Figure 3.



Project name: WordCount

☑ Use default location

Location: /home/cloudera/workspace/WordCount

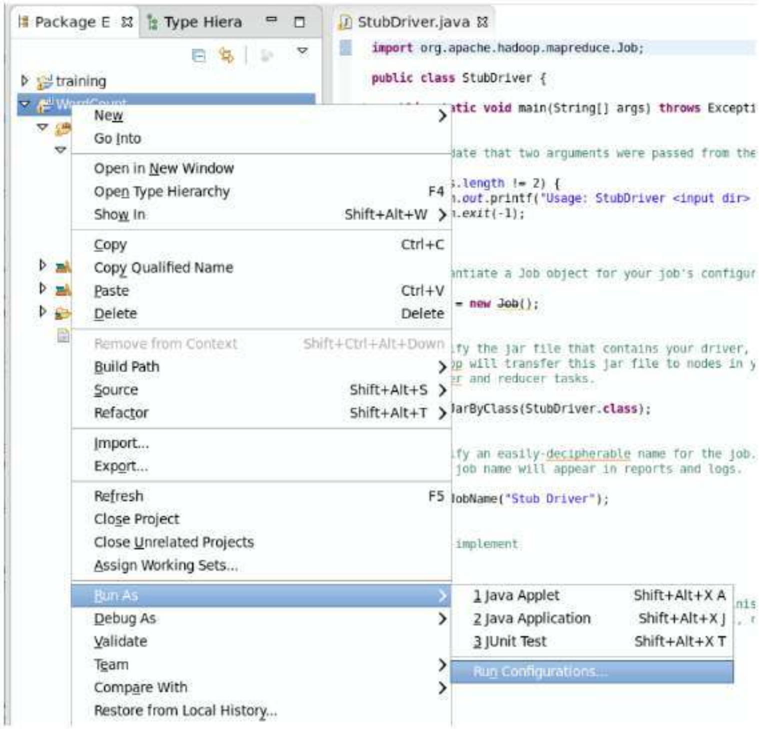
Figure 3: Create a Hadoop Project.

5. Modify or replace the stub classes found in the `src` directory as needed.

2.2 Running Hadoop jobs in standalone mode

Once you've created your project and written the source code, to run the project in stand-alone mode, do the following:

1. Right-click on the project and select *Run As* → *Run Configurations*. See Figure 4.



```
import org.apache.hadoop.mapreduce.Job;

public class StubDriver {
```

Figure 4: Run a Hadoop Project.