CS 4593/5463 Cloud and Big Data

Assignment 2: Hadoop Cluster Setup, and MapReduce Programming
Due Midnight Monday, Mar 6, 2017

1. Download the latest/updated Lecture 4(c) from the blackboard, and follow the instructions on slide 22-34 to setup a hadoop cluster in OpenStack cloud. The cluster should have 1 master, and 3 worker nodes.

2. Download the Austin Police reports (2008-2011) from the data.gov site.

```
wget -O apd08.csv https://data.austintexas.gov/api/views/r6sg-xka2/rows.csv?accessType=DOWNLOAD

wget -O apd09.csv https://data.austintexas.gov/api/views/ei2n-fehk/rows.csv?accessType=DOWNLOAD

wget -O apd10.csv https://data.austintexas.gov/api/views/4c6h-tv2y/rows.csv?accessType=DOWNLOAD

wget -O apd11.csv https://data.austintexas.gov/api/views/gr59-ids7/rows.csv?accessType=DOWNLOAD
```

Incident Report Number	Crime Type	Date	Time	LOCATION_TYPE	ADDRESS
2010520382	DEADLY CONDUCT	2/21/2010	248		600 BLOCK W WILLIAM CANNON DR
20101420417	PUBLIC INTOXICATION	5/22/2010	255		300 BLOCK E 6TH ST
2010911514	PUBLIC INTOXICATION	4/1/2010	1604		3600 BLOCK DUVAL RD
2010842386	DWI	3/25/2010	2338		1000 BLOCK W 6TH ST
20102250200	PUBLIC INTOXICATION	8/13/2010	204		600 BLOCK NECHES ST
20105044361	THEFT OF BICYCLE	8/21/2010	1636		1000 BLOCK JUSTIN LN
20105066208	BURGLARY OF VEHICLE	12/14/2010	1330		1900 BLOCK FAIRLAWN LN
20103022364	CUSTODY ARREST TRAFFIC WARR	10/29/2010	2252		800 BLOCK N IH 35 SVRD SB
20101691220	VIOL CITY ORDINANCE - DOG	6/18/2010	1345		500 BLOCK W 12TH ST

3. Create an input directory in HDFS, and copy the downloaded Austin police reports to HDFS.

```
cd $HADOOP_PREFIX
bin/hadoop fs -mkdir /hw2-input
bin/hadoop fs -copyFromLocal ~/*.csv /hw2-input/
```

4. Write a MapReduce program (in Python) that will answer the following

   Where is most of the crime happening in Austin? What types of Crime are happening in that location?

5. Write a MapReduce program (in Python) that will construct a co-occurance matrix between crime type, and month of the year.

   Example,

   DEADLY CONDUCT 10 20 1 0 4 5 9 34 7 1 11 0
   PUBLIC INTOXICATION 20 50 10 0 4 15 9 4 7 1 12 0
   …

6. **CS 5463 Graduate Students/ Extra Credit for CS 4593:** Find out at least two Hadoop configuration parameters that can be tuned to improve the performance of your MapReduce programs. Compare the job execution times with and without parameter tuning.

Important Note:

Job execution results can be obtained by using the following command

```
bin/hadoop job –history <output-directory>
```

Here, the job output directory is used to identify the job, whose execution history is being fetched. Alternatively, job history can also be obtained from the jobtracker web interface:

http://129.115.xx.xx:50030

Troubleshooting Tips:

(a) If a job is long running, you can let it run in the background, and free the shell to do other stuff by using:

```
Ctrl-C
```

(b) Job progress can be monitored from the Jobtracker Web Interface. You can find out which task is taking too long or which failed.

(c) If a job hangs (making no progress), you can kill the job as follows:

```
bin/hadoop job –list

bin/hadoop job –kill job_2014----
```

(d) To troubleshoot the task that took too long, check the corresponding log file under the directory,

```
/usr/local/hadoop-1.2.1/logs/userlogs
```

Or,

Check the Log files from the jobtracker web interface.

```
http://10.242.144.xx:50030/logs/userlogs
```

Submission Policy and Deliverables

Only one submission per group is required. Submission should include the following.

1. MapReduce programs (python files)
2. The resulting output files of your MapReduce programs
3. A PDF report that includes:

    a. A graph comparing the performance of your MapReduce job with and without parameter tuning.
    b. One representative Screenshot of the console output when you execute the benchmark.
    c. One representative Screenshot of the Jobtracker's web interface which shows the status of running/completed jobs.
    d. Describe how the work was divided among your group members.