Week 3

Configuring RHadoop on Rustler

**Configure HADOOP\_HOME**

hadoop version

export HADOOP\_HOME=/usr/lib/hadoop

**Configure JAVA\_HOME**

locate /bin/java

export JAVA\_HOME=/usr/java/jdk1.7.0\_67

**Configure HADOOP\_STREAMING**

locate hadoop-streaming-

export HADOOP\_STREAMING=/usr/lib/hadoop-mapreduce/hadoop-streaming-2.3.0-cdh5.1.0.jar

**Download datasource**

wget <http://www.gutenberg.org/cache/epub/48276/pg48276.txt>

**>> Wordcount.R**

#!/usr/bin/env Rscript

library(rmr2)

bp = rmr.options("backend.parameters");

#bp$hadoop[1] = "mapreduce.map.java.opts=-Xmx1024M";

#bp$hadoop[2] = "mapreduce.reduce.java.opts=-Xmx512M";

bp$hadoop[1] = "mapred.tasktracker.map.tasks.maximum=1";

bp$hadoop[2] = "mapred.tasktracker.reduce.tasks.maximum=1";

#bp$hadoop[3] = "mapreduce.map.memory.mb=1280";

#bp$hadoop[4] = "mapreduce.reduce.memory.mb=2560";

rmr.options(backend.parameters = bp);

rmr.options("backend.parameters")

wordcount =

function(

input,

output = NULL,

pattern = " "){

wc.map =

function(., lines) {

keyval(

unlist(

strsplit(

x = lines,

split = pattern)),

1)}

wc.reduce =

function(word, counts ) {

keyval(word, sum(counts))}

mapreduce(

input = input ,

output = output,

input.format = "text",

map = wc.map,

reduce = wc.reduce,

combine = T)}

from.dfs(wordcount("book.txt"))

**Executing**

chmod +x wordcount.R

./wordcount

**Datasets**<http://en.wikipedia.org/wiki/Wikipedia:Database_download>  
wget http://dumps.wikimedia.org/enwiki/latest/enwiki-latest-abstract1.xml

<http://www.kdnuggets.com/datasets/index.html>

**SparkR on Rustler**

**https://github.com/amplab-extras/SparkR-pkg**

% git clone https://github.com/amplab-extras/SparkR-pkg

% cd SparkR-pkg/

% hadoop version

>>Subversion git://github.sf.cloudera.com/CDH/cdh.git -r >>8e266e052e423af592871e2dfe09d54c03f6a0e8

>>Compiled by jenkins on 2014-07-12T13:49Z

>>Compiled with protoc 2.5.0

>>From source with checksum 7ec68264497939dee7ab5b91250cbd9

>>This command was run using /usr/lib/hadoop/hadoop-common-2.3.0-cdh5.1.0.jar

% SPARK\_HADOOP\_VERSION=2.3.0-cdh5.1.0 ./install-dev.sh

**Run a sample SparkR job**

./sparkR examples/pi.R "local" 100

**RHIPE ?**