Home Work 3 Problem 1

a)

- ADRIANO: 111
- Whether : 41
- love : 2221
- loves : 203
- the : 25578
- whether : 79
- we : 2922
- zodiac : 1

b)

First we need to download the file to our local computer, then:
The command: ***$ tr ' ' '\n' < part-r-00000 | sort -u | wc -l***

Or

$ wc –l part-r-00000

The number of different words: 29183 (including some special words, such as: 1; 10; 2; 2d ect.)

c)

Suggestion 1:

We can only count positive words such as: love, loves, like, happy; as well as the negative words such as: dislike, hate, hates. Then we label the “positive” words and the “negative” words, and sort them into different level, such as hate-Level One (completely) and dislike-Level Two (medium).

Suggestion 2:

It will save time and space to ignore the words have nothing to do with sentiment analysis. We can ignore some “useless” nouns such as person’s name, address. It will also save a lot of time.

d)

Record Reader divides the whole file such as articles into different small pieces and also matches them

Input data are HDFS files, output data are <key, value> pairs, keys are numbers represent the amount of characters, values are some words in a sentence which can be used by Mapper.

e)

The folder of word counts contains two files: SUCCESS and part -r-00000. The file SUCCESS corresponds to the MapReduce runs successfully, the file part-r-00000 corresponds to the output of MapReduce. Reducer is responsible for those files.

Wustl key: wushuo student ID: 452666

Home Work 3 Problem 2

a)

There will be significant skew.
Since there are 200 nodes and with various reduce() functions which equal to various reduce tasks, so we could execute reach reduce task at a different node. There is often significant variation in the lengths of the value lists for different keys, so different reducers take different amounts of time. If we make various reduce functions process their value list, then the tasks themselves will exhibit skew.

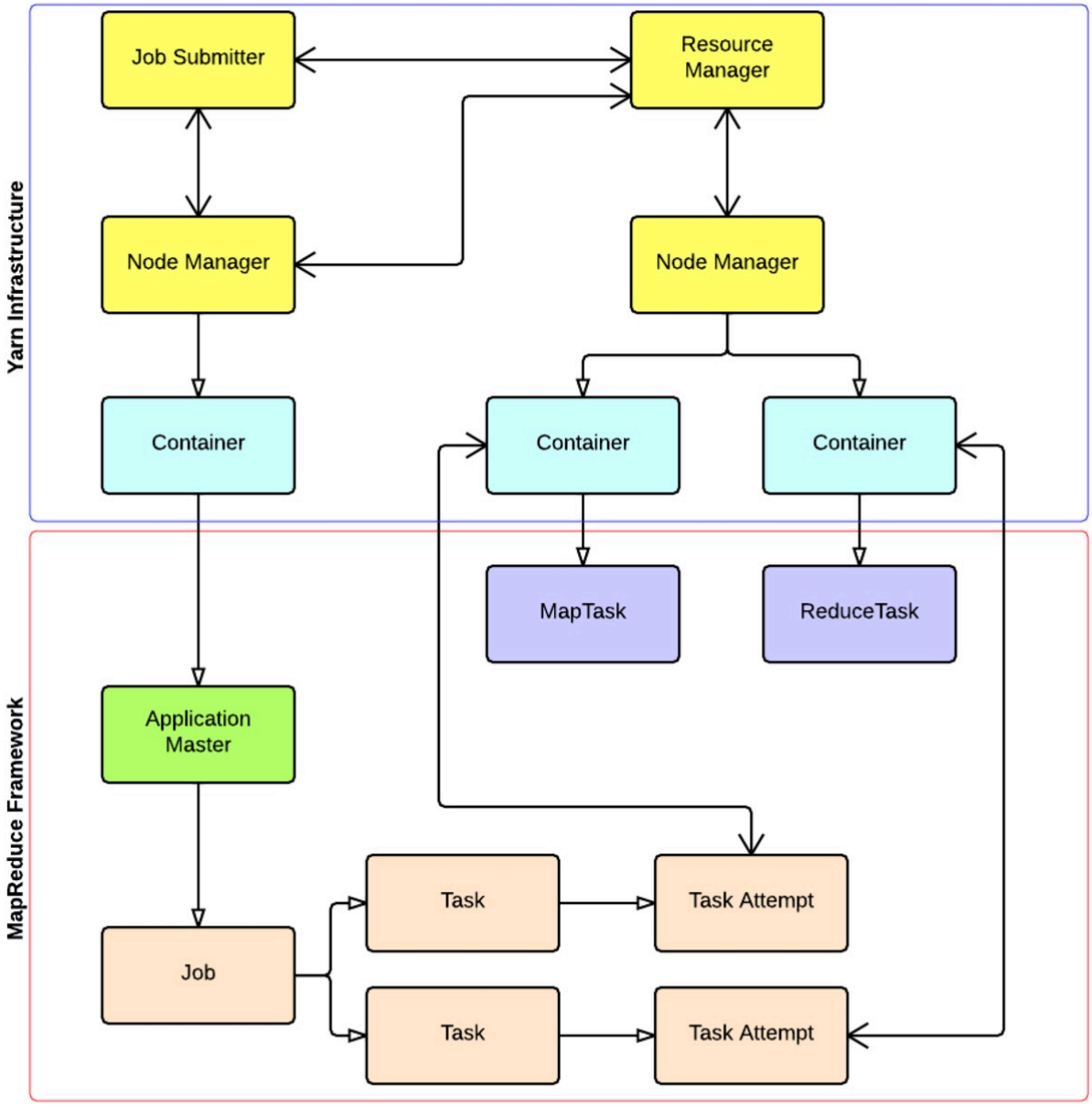
b)

10 tasks: no signification skew,
Since We can reduce the impact of skew by using fewer Reduce tasks than there are reducers. If keys are sent randomly to Reduce tasks, we can expect that there will be some averaging of the total time required by the different Reduce tasks. Therefore, there is no signification skew.

10,000 tasks: no signification skew,
Since tasks A = tasks B+C+D= will take 100 second. Then, we can further reduce the skew by using more Reduce tasks(10,000) than there are compute nodes(200). In that way, long Reduce tasks might occupy a compute node fully, while several shorter Reduce tasks might run sequentially at a single compute node. For example, we can put A on one node, and B&C&D on another node.
Therefore, there is no signification skew too.

Home Work 3 Problem 3

a)



4 types of daemons:

- Resource Manager: job scheduling
- Application Master: task progress monitoring
- Node Manager: run tasks and send progress reports
- Job History: archives job metrics and metadata

Job execution:

- Put data in HDFS (data nodes)
- Execute driver on client (submit job to the resource manager)
- Resource manager invokes application master
- Application master invokes node manages for task execution

Input data:

- When possible: Computer nodes
- If not possible: Map task transfers the data across the network from HDFS

Output data: local disk

Intermediate data: is transferred across the network, no data locality

b) The data locality optimization in Hadoop is that the computation tasks are moved to the nodes that hold the data, or if not there, nodes that are have a fast network path to the nodes that hold the data. It's common to see 90% of the network IOs being local, 8% being rack-local, and 2% being remote. (Your balance may vary.)

It is applicable for map phases and reduce phases.

If tasks can not be executed local to data, the map tasks will transfer the data across the network. What's more map tasks can't store their output on local disk. Since reduce tasks need to wait for map tasks to have finished, so there will be a bottleneck.