REPORT

Yaxin Huang

02/01/2015

This report is provided to answer the questions below:

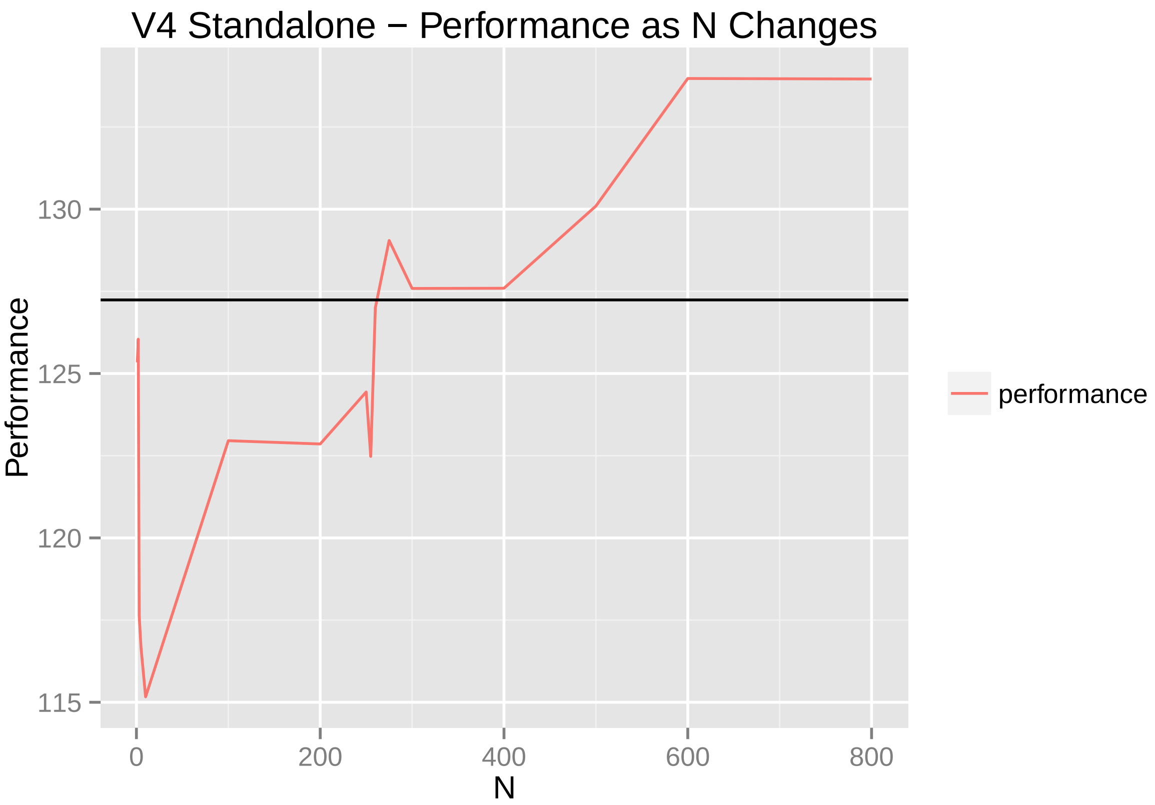
(Q1) What is the performance difference between v1, v2 and v3 where you run v2 and v3 both in a single Hadoop process and "pseudo" distributed processes.

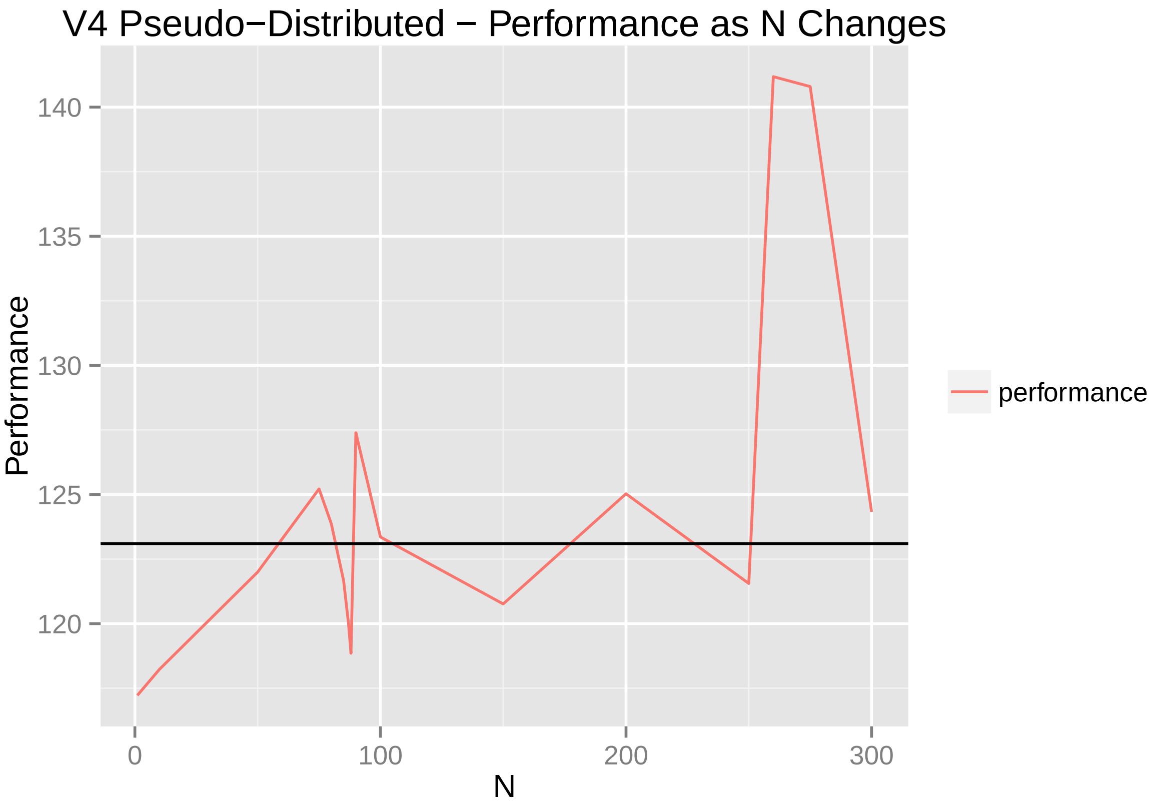
Version	Standalone	Pseudo-Distributed
1	333.44s	N/A
2	127.239s	123.099s
3	682.51s	715.64s

The performance difference among version 1,2,3 is displayed in the above table. Version 2 is almost three times as efficient as version 1, which is probably because version 2 is implemented by using MapReduce. In the pseudo-distributed mode, because I was using single node, the performance is almost the same as using standalone mode.

Version 3 is much slower because I don't know how to deal with the composite key. When the Mapper function uses (Category,Sale) as the composite key and pass it to Reducer, each call to the reducer function can only catch one pair of them. I couldn't figure out what to do, since I have no way to compute the median in this case. My solution adds another job which the previous Reducer output the composite keys to an intermediate file and the second job will do the remaining work. But this is not why we want to use composite key – I've tried using Combiner but it didn't work, so I really wonder how others solve this problem.

(Q2) Comparing v(2|3) with v4, what is the largest value of N that does not affect performance?
Because my version 3 is far less efficient than version 2, here I use versoin 2 as the counterpart.





In the first figure, the black line shows the average performance of version 2 in standalone mode, 127.239s. It is clear in this graph that when N is around 255 to 260, the calculation of Fibonnaci of N starts to affect the performance.

In the second figure, since I'm also running other programs, the performance varies a lot. But as the graph shows, when N is larger than 250, the performance starts to go up. Thus, in the pseudo-distributed mode, when N comes to 250, the performance will be affected by the Fibonacci calculation.

(Q3) How many instances of the reducer are running?
Because both in the standalone mode and pseudo-distributed mode, I didn't set the number of reducer used in the main class, the default number is used, which should be 1.