Machine Learning for Large Data Sets - Assignment 1
Multi-label Multi-Class classification using Naive Bayes with Map Reduce

D Ravi Shankar [*1]

We have to classify multi label multi class data using naive bayes on both local system and using map-reduce frame work with hadoop.

1. Local System

Local System results: Implemented by iterating through the training data and formed a matrix of size : rows = no.of unique words (256278), columns = no.of labels (50).

Training set accuracy: 86.48
Development set accuracy: 82.21
Test set accuracy: 81.24

Time taken for training: 551 seconds
Time taken for testing: 47 seconds

2. Map Reduce framework

Map Reduce: 5 map reducers are used. 1st map-reduce for label counting, 2nd map-reduce for word counting, 3rd map-reduce for getting words with line numbers and labels, 4th map-reduce for joining the word counts and word line numbers and 5th one for calculating the probabilities and predicting labels line wise. First two map-reduces are required only while training. For testing the next 3 map-reducers are used.

Training set accuracy: 83.16
Development set accuracy: 79.04
Test set accuracy: 67.74

Time taken for Training (with 2 reducers) : 192.2 seconds
Time taken for Testing (with 2 reducers) : 474 seconds

3. Observations

1) As the number of reducers are increased the time taken decreases approximately exponentially. 2) Compared to local system implementation, the map-reduce framework takes much lesser time for training.The reason for this is in local system, we have to iterate through the entire data sequentially and then update the counts. But in map-reduce framework we can do these counts in parallel.

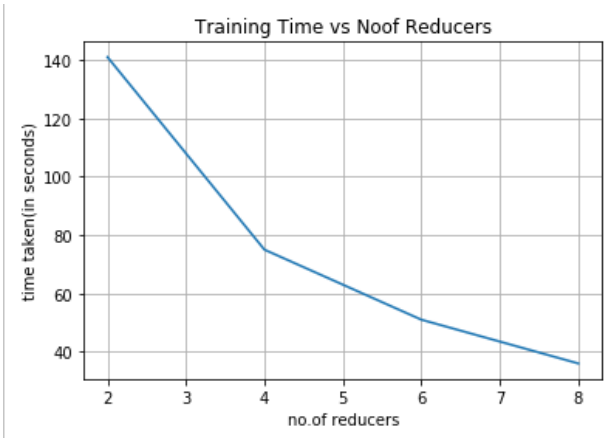

Figure 1. Training Time taken vs No.of Reducers

But it takes much more time for testing(for lesser no.of reducers, as we increase no.of reducers, the total time taken by map-reduce framework is significantly lesser).
The reason for this is in local system after training we have a matrix with all the counts. While testing we use it as a lookup table, In map-reduce framework, there is no such table. We have to add the counts sequentially. So as the no.of reducers are increased, this can be done in parallel.

Total time with 8 reducers: 209 seconds (65 training + 144 testing)

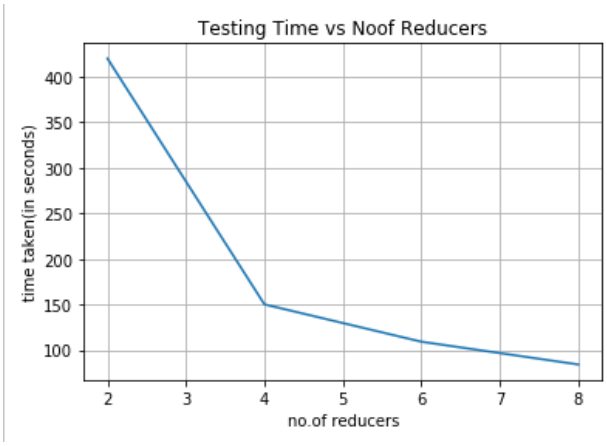

Figure 2. Testing Time taken vs No.of Reducers