Technical Report for Third Project in Parallel Computing

Yasheng Sun
117020910076
sunyasheng123@gmail.com

Abstract

This report concludes the third project in parallel computing. This projcet target at implementing word counting pragram using MapReduce. This is a pretty elementry task in MapReduce Framework, which follows the classical MapReduce Pattern-Summarization Pattern. We will explain whole workflow briefly in this report. All my code is publicly available on my github https://github.com/sunyasheng/Parallel-Computing.

1. Data Preparation

In this project, we choose the novel Gone with The wind as the system input as is shown in Fig. 1. This novel contains 2657422 characters including Chinese, English and Punctuation. Mixture of various types of languages and characters aims to mimic the true scenario in real world.

2. Methodology

In this section, the whole workflow of MapReduce is illustrated in Fig. 2. The task of Mapper is to select every English item from orignal article and map the corresponding item to 1. Then the output stream will be shuffled, sorted and passed to Reducer. Reducer will go through all the items and add up all the numbers for each item.

Note that all English items have to be transported from the Mapper Node to Reducer Node, which implies that if a certain English item will be transported mutiple times if it is repeated in an article many times. The inter-node transportation is usually very expensive. To lower the amount of transportation, A practical technique Combiner is introduced here to add up the number of items in its current Mapper Node. The Combiner is essentially a lcoal reducer. With this pre-reduction, the transportation amount between different nodes can be decreased significantly.

3. Result

Only part output of this mapreduce task is shown in Fig. 3 because there are too many words to be completely presented here. The word and their frequency of exisitence are listed row by row.

From Fig. 4 we could know that the mapper task is split to two nodes and the output stream will be passed to one reduce node to finally summarize the result after map operation.

Fig. 5 shows that the Combiner technique significantly reduces the amount of reduce input records and reduce shuffle bytes compared with the traditional routine without

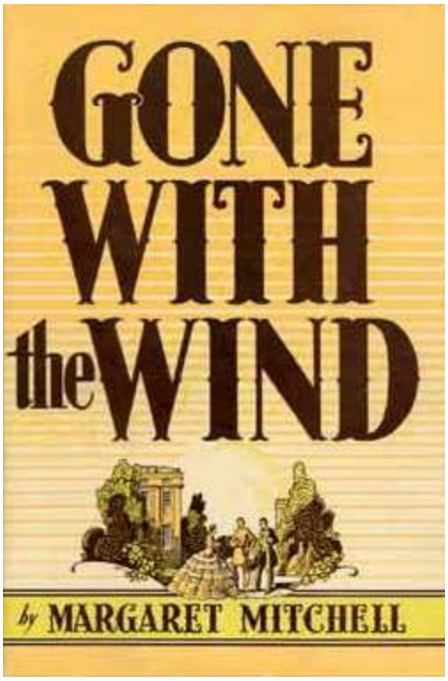

Figure 1. Gone with the Wind

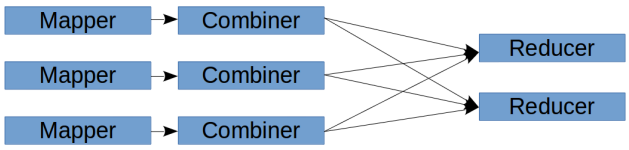

Figure 2. The MapReduce workflow.

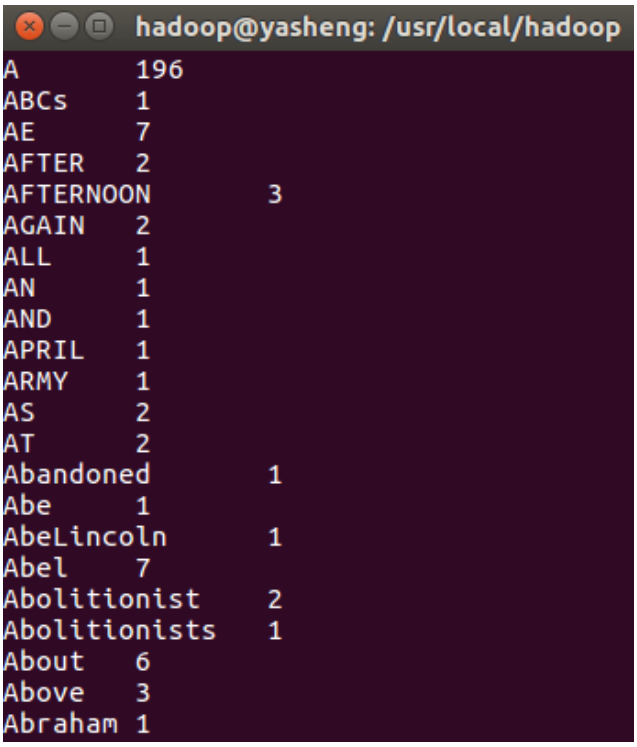

Figure 3. System output for word count program.

Task Type	Total
Map	2
Reduce	1

Figure 4. System information in MapReduce Task.

Reduce input groups	0	30,118
Reduce input records	0	38,808
Reduce output records	0	30,118
Reduce shuffle bytes	0	504,131

Figure 5. The reduce input records and shuffle bytes information with Combiner.

Combiner as is shwon in Fig. 6. This improvement will play a more important role when it comes to a larger scale dataset.

4. Conclusion

A sample word count program is presented in this project, by which we show the MapReduce paradigme – Summarization Pattern. A practical technique, Combiner, is introduced here to lower the transportation amount between Mapper Node and Reducer Node. Future work will involve more complicated problems in large scale.

Reduce input groups	0	30,118
Reduce input records	0	402,832
Reduce output records	0	30,118
Reduce shuffle bytes	0	4,166,472

Figure 6. The reduce input records and shuffle bytes information without Combiner.

References

[1] http://sunyasheng.github.io/