MRPack Manual

Brief Description:

Hadoop based Concurrent algorithms in a single MapReduce Job: MRPack is a variant of MapReduce implementation where multiple related algorithms can be executed concurrently in a single MapReduce job. In MRPack, All algorithms are implemented in Map and Reduce programming model and all these algorithms are then executed under single unified Map and Reduce super class. MRPack uses Main Map and reduce, and Sub map and reduce approach to implementation. It introduces a composite key structure for intermediate data handling.

The Code Structure:

MRPack code contains two main directories IMIR and main. IMIR directory contains the implementations of Mappers, Reducer and other classes and methods for each algorithms including InvertedIndex, WordCount, KNN, KMeans. Each algorithm is a sub-directory of IMIR. The main directory contains the Main MapReduce job implementation and the mappers/reducers of the main job. The main Mapper and reducer have references to the algorithm mapper/reducer in IMIR directory.

How to Change:

To change or modify IMIR, One need to write the specified algorithm code separately as in IMIR we have codes for different algorithms. And then create reference for that new algorithm in the main Mappers and reducers.

Similarly, to change existing algorithm code, change should be made directly to the IMIR sub-directory codes consisting algorithms.

The input and output can only be processed and changed in the main directory i.e. Main Job code of MRPack.

How to Execute:

MRPack on Eclipse

Install Hadoop and

1. Install eclipse and configure with Hadoop, if you want to use eclipse
2. Open MRPack in Mapreduce Perspective
3. Set the arguments in Eclipse : Input directiry for data and output directory at Hadoop
4. Execute as an Hadoop Job. (Run on Hadoop option)

MRPack asd an executable Jar

Install Hadoop and then

1. Creat Jar file from MRPack in Eclipse by "Export as jar" option. Remember to set the main class of the jar.
2. Execute the Jar file on Ubuntu terminal followed by input data directory and output data directory in Hadoop.
    - For example:
    - hdfs jar MRPack.jar /home/*****/*hduser/input/data /home/*****/hduser/output/data

We have implemented for the data to be simple text format. If your data format changes, then you better change only implementation of the Main mapper function of the main job.

For any queries:

Email: idris@oslab.khu.ac.kr