Assignment 4 of CISC 3018

ZHANG HUAKANG

May 17, 2021

1

1.1

Advantage

- Efficient: It decompose the huge task into smale pieces which can be processed in parallel.
- Scalable: Flexibly scale up/down by controlling the number of machines used by MapReduce.
- Fault-tolerant: Robust when some of the machines fail

Disadvantage Since the MacReduce will store the intermediate files and the outputs in the disk(HDFS in fact), when we run the MapReduce in sequence, it will read and write the data in local disk which will cost many time and lead to a low-efficiency.

1.2

Similarity They both have similar architecture: the master node distributes tasks and the worker node processes the task in parallel.

Difference The MapReduce will store the intermediate files and the outputs in the disk(HDFS in fact) while the Spark stores those data in memory which lead a faster reading and writing speed in the repeating process.

2

For narrow RDD, each parent RDD will only generate a child RDD. For wide RDD, the parent RDD will generate multiple child RDDs. Thus, narrow RDD has less redundance which wide RDD has more reliability.

3

3.1

- Batch processing: process a significant amount of data in a parallel and distributed manner. No strict latency limit to complete jobs.
- Streaming: Data is generated in real-time fashion as the time passes, which needs to be processed when they arrive in sequence as a 'stream'. Processing data/tasks as 'data flows'
- Interactive processing: Provide feedback results instantly when a query is posed

3.2

- Batch processing: MapReduce.
- Streaming: Apache Storm, and Twitter Huron.

4

4.1

- Decision Tree based Methods
- Support Vector Machines
- Neural Networks

4.2

- Decision Tree based Method: Use the generated decision tree for deterring the 'label' of a new data
- SVM based Method: find a hyperplane to separate two groups of nodes in space.

5

5.1

The train is the process to find the parameters of each node in network, optimize those parameters in order to get a more accurate results.

5.2

For one node N_a in hidden layer, the output

$$y_a = f(\sum_{i=1}^{n} w_i x_i + b_a)$$

where $x_i, i \in [1, n] \cap \mathbb{N}^*$ is outputs, w_i and b_a is the parameters of node N_a and f is activation function. y_a will be a output of all nodes in next layer.

5.3

- Classification: Before the model training, we have already known about the number and type of labels.
- Cluster: Only when we finish the process, we will know how it groups the data.