Wyatt Blair

EMT-678-WS: Big Data Technologies

9/19/2024

**Class 2—** **Google MapReduce-OSDI 04**

This reading, authored by two Google engineers, discusses the MapReduce protocol which allows for actions to be taken on a dataset without bogging down the system resources too greatly. Algorithmically, MapReduce needs the user to define two functions prior to use: the map function and the reduce function. The map function accepts a key and a value, and then uses this pair to produce a list of intermediary key and value pairs. Then, the reduce function uses a single intermediary key to collect all values associated with that key and performs some function on this information to create an aggregated list of values. This list of values and the intermediary key will be presented as output to the user. The authors provide some common uses for MapReduce including: Distributed Grep, Count of URL Frequency, Reverse Web-Link Graph, Term-Vector per Host, Inverted Index, and Distributed Sort. In each of these examples, the authors demonstrate how the problem can be viewed through the lens of MapReduce to create a scalable solution.

The scalability of MapReduce comes from its distributed nature. A large dataset can first be split into even chunks. A designated set of automated “workers“ can run the map function on each chunk concurrently. As these workers complete their map function execution, another set of workers will complete the reduce portion of the function. Not only does this increase the speed of large data operations (since the computation is being split among any number of separate computers) but it also increases stability. If a particular worker encounters an error with their chunk MapReduce can simply continue with the other workers without catastrophic error.

All-in-all MapReduce is a methodological approach to operations on a large dataset which allows for future scalability and proper distribution of the computational burden on available resources.