BIG DATA PAPER SUMMARY

Jeffrey Lupia

10/19/2016

- MapReduce: Simplified Data Processing on Large Clusters

  - Jeffrey Dean and Sanjay Ghemawat jeff@google.com, sanjay@google.com Google, Inc.

- A Comparison of Approaches to Large-Scale Data Analysis

  -Andrew Pavlo, Erik Paulson, Alexander Rasin, Daniel J. Abadi, David J. DeWitt, Samuel Madden, Michael Stonebreaker

- ICDE 2015 Talk by Michael Stonebreaker discussing his “10 Year Test of Time” Paper award

MapReduce: Simplified Data Processing on Large Clusters

- This paper describes the programming model of MapReduce, a method by which large sets of data are able to be broken down using parallelization. They can then be computed using clustered computing, by allocating program tasks to a series (or cluster) of less powerful, and in turn less expensive machines.
- This method allows huge sets of data to be processed in a relatively cheap way.
- Using this method, tasks can be abstracted into two parts, the map and reduce.
- The map function takes in a user input as a set of user-defined key values as pairs and maps them into a separate set of intermediate pairs for the purpose of the reduce function.
- The reduce function then takes these intermediate values and further breaks them down into another set of result data pairs.
- The map, and reduce functions can be easily manipulated by the user, and has a wide variety of computing applications.

Implementation

- The MapReduce functions can be implemented widely across many industries and uses.
- Because the user is able to define the map and reduce functions, as well as being able to easily manipulate the input, the applications of these functions widespread.
- It is often used by companies to track and process user inputs, as well as having the ability to quickly sort through large, and numerous documents. It can give data about the frequency of words, or can be used to sort through financial records, or inventories in which the amount of data is immense.

Analysis of MapReduce Paper

- MapReduce seems to be a very versatile and cost effective way to sort through large quantities of data, which would otherwise take too much time, and energy to process.
- Because of the user defined *Map* and *Reduce* functions, the MapReduce method of big data processing has extremely widespread uses throughout many disciplines and in many different contexts.
- This method seems to be very useful for data programmers, as in using it, users will have a much easier time writing code, as the parallelization aspect of the code is taken care of.
- This method seems to be also very safe, regarding the integrity of the processing, as when a worker program processing the data fails, the data is not dropped, but is rather designated for reprocessing.

A Comparison of Approaches to Large-Scale Data Analysis

- This paper compares several different types of commonly used methods to process large amounts of data, specifically looking at clustered computing, and discusses the advantages and shortcomings of each method.
- Specifically the four methods looked at by this paper are Hadoop, MapReduce, Vertica (a parallel DBMS), and DBMS-X
- The general idea is to determine what the most effective and versatile option for large data processing and analysis is.

Implementation

- This comparison was made by putting each system through various tasks, and measuring their results.
- These tasks included:
  - The “Grep Task” described in the MapReduce Paper
  - Aggregation task
  - Join task
  - Selection task
- The tasks were assessed based on data loading times and task execution times

Analysis of Comparison Paper

- The paper highlighted the different strengths of various methods for large data analysis.
- As it turned out, the MapReduce method was much more easily implemented, as the process to load data and prepare it to be executed was easier than the others.
- The DBMS's however, were noticably more efficient in the actual execution of the data.
- I feel that aspects of both systems should be improved upon moving forward technologically to provide the most efficient data processing.

Comparison of the Papers

- The MapReduce paper was very comprehensive in it's description of the technical details working in the MapReduce Method. It was far more technical in its language, however also gave very specific descriptions of the implementations and uses of the system.
- The MapReduce paper focused very much on how the system was very versatile in its uses because of the user-defined *Map* and *Reduce* functions, as well as the adaptable user input.
- The second paper recognized the MapReduces versatility and the qualities that made the programming easier for the user, however determined that as far as actual performance goes, MapReduce was not the Most efficient, or fastest-processing method available.

Main Idea of Stonebraker Talk

- The Stonebraker talk focused on how one size does not fit all, and that problems faced in the technological sector are ever-changing and are constantly being reevaluated and re-solved with new and inventive solutions
- He discussed how older data modeling systems are becoming outdated. And asserted that soon, many of these options will be phased out by newer models, specifically, he mentioned Row-stores.
- Stonebraker, however believes the future of data processing and modeling will be the use of column-stores.

Advantages and Disadvantages of MapReduce

- MapReduce seems to be a viable option, but as technology moves forward, it will need to evolve to be faster, and more competitive in a market which is always evolving with new solutions.
- The comparison paper showed that other DBMS's were faster than MapReduce, however, MapReduce was unique in its ease of use, and versatility of applications.
- Moving forward, I think that MapReduce can be the prominent system of Large data Processing and storage if it is improved upon in its speed of task execution and processing data.