High Performance Clustering of Social Images in a Map-Collective Programming Model

Bingjing Zhang, Judy Qiu
Department of Computer Science, Indiana University Bloomington

Abstract

Large-scale iterative computations are common in many important data mining and machine learning algorithms. Most of these applications can be specified as iterations of MapReduce computations, leading to the Iterative MapReduce programming model [1] for efficient execution of data-intensive iterative computations interoperably between HPC and cloud environments. We observe that a systematic approach to collective communication is essential but notably missing in the current model. Thus we generalize the iterative MapReduce concept to Map-Collective on the premise that large collectives are a distinctive feature of data intensive and data mining applications. To show the necessity of Map-Collective model, this paper studies the implications of large-scale social image clustering problems, where 10-100 million images represented as points in a high dimensional (up to 2048) vector space are required to be divided into 1-10 million clusters.

Categories and Subject Descriptors

C.2.4 [**Computer-Communication Networks**]: Distributed Systems – *Distributed applications.*

Keywords

Social Images, Data Intensive, High Dimension, Iterative MapReduce, Collective Communication

1 Major Results

Data mining dominated by collectives with large size 512 MB messages requires new technologies. Our new broadcast collective is four times faster than the best Java MPI and gives 20% better performance than the fastest C/C++ MPI methods, in addition to factor of 5 improvement over a non-optimized (for topology) pipeline-based method on 150 nodes (see Figure 1). Our new algorithm scales much better than Spark [2]. Local aggregation in Map stage reduces the size of 20 TB intermediate data by at least 90%. These communication improvements will be combined with triangle inequality optimization [3] (Elkan's algorithm [4] extended for large problem sizes). Different optimizations for Azure give Twister4Azure [5] better performance than current MapReduce and Iterative MapReduce Azure platforms.

2 Experiment

To execute an image clustering application with 7 million image feature vectors and 1 million clusters, we use 10,000 Map tasks (125 nodes, each of which has 8 cores). In broadcasting, the root node broadcasts 512 MB of cluster center data to all nodes. Thus the overhead of a sequential broadcasting is substantial.

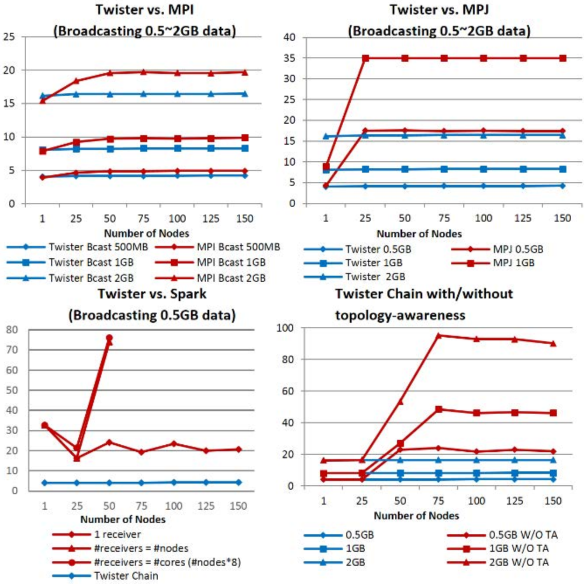

Figure 1. Performance comparisons of broadcasting methods

3 Novel Collective

We propose a topology-aware pipeline-based chain method to accelerate broadcasting for the Iterative MapReduce model. More details can be found at [6].

4 References

[1] J. Ekanayake, H.Li, B. Zhang, T. Gunarathne, S. Bae, J. Qiu, G. Fox. “Twister: A Runtime for iterative MapReduce.” The First International Workshop on MapReduce and its Applications of ACM HPDC 2010 conference, June 20-25, 2010. 2010, ACM: Chicago, Illinois.

[2] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica. “Spark: Cluster Computing with Working Sets.” *HotCloud*, 2010.

[3] J. Qiu, B. Zhang, “Mammoth Data in the Cloud: Clustering Social Images.” *Clouds, Grids and Big Data*, IOS Press, 2013.

[4] Charles Elkan, “Using the triangle inequality to accelerate k-means.” *Intl. Conf. on Machine Learning* 2003.

[5] T. Gunarathne, B. Zhang, T.-L. Wu, and J. Qiu. “Scalable Parallel Computing on Clouds Using Twister4Azure Iterative MapReduce.” *Future Generation Computer Systems* (29), pp. 1035-1048, 2013.

[6] B. Zhang, J. Qiu. High Performance Clustering of Social Images in a Map-Collective Programming Model. Technical Report, July 2, 2013.