University of Zurich[UZH]

Department of Informatics

University of Zürich
Department of Informatics
Binzmühlestr. 14
CH-8050 Zürich
Phone. +41 44 635 43 11
Fax +41 44 635 68 09
www.ifi.uzh.ch/dbtg

UZH, Dept. of Informatics, Binzmühlestr. 14, CH-8050 Zürich

Thomas Brenner

Prof. Dr. Michael Böhlen
Professor
Phone +41 44 635 43 33
Fax +41 44 635 68 09
boehlen@ifi.uzh.ch

Zürich, May 14, 2013

BSc Thesis
Datenbanktechnologie

Topic: Load Balancing Implementation in Hadoop

Hadoop is a MapReduce framework. It splits a huge chunk of data into smaller blocks. Each block is processed individually by a mapper job, having multiple mappers running in parallel. The role of a mapper is to transform the input data into an intermediate set of data in a <key, value> format. Output from mappers are then grouped together in clusters, each cluster having all pairs sharing the same key. Multiple clusters are combined together to form a partition. Eventually, a partition is passed on to a reducer to perform further computations and produce the end result.

Hadoop is responsible for creating such partitions. However it does not guarantee even-sized partitions. It is possible that it creates a partition containing 100 entries, and another partition with only 5. When these two partitions are given to their corresponding two reducers, it will be expected that the reducer with the larger partition will require more execution time. The goal is to implement a load balancing algorithm in hadoop to reduce the variation between the partition sizes.

TopCluster is a distributed monitoring approach for MapReduce systems that computes a global histogram of (key,cardinality) pairs, which approximates the cardinalities of the clusters with the most frequent keys [1]. In this project, you will study the TopCluster algorithm and implement it inside Hadoop. As use case, implement the constant interval extraction from temporal databases.

Tasks

1. Study the TopCluster approach that computes the global histogram of the data [1].
2. Work out an example using constant intervals, illustrating the local and global histograms produced.
3. Implement local and global histograms inside Hadoop.
    - Compute local histogram per mapper
    - Transmit histogram from mapper to job tracker
    - Compute global histogram based on received local histograms
    - Distribute partitions over reducers based on the global histogram
4. Write thesis, covering the following aspects:
    - explanation of the TopCluster approach,
    - challenging aspects encountered during implementing TopCluster into Hadoop,
    - cases when TopCluster is most efficient, and is least efficient with respect to constant intervals,
    - experimental results.

Supervisor: Amr Noureldin (noureldin@ifi.uzh.ch)
Start date: 25 April 2013
End date: 20 August 2013

University of Zürich
Department of Informatics

Prof. Dr. Michael Böhlen

References

[1] Benjamin Gufler, Nikolaus Augsten, Angelika Reiser, and Alfons Kemper. Load balancing in mapreduce based on scalable cardinality estimates. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*, ICDE '12, pages 522–533, Washington, DC, USA, 2012. IEEE Computer Society.