GPU acceleration of compute-intensive applications has emerged as a new research frontier with phenomenal success-rates. Such applications are characterized by large data-sets being processed by singular functional units (FUs) often described as SIMD (Single Instruction Multiple Data) computing. Moreover, with the proliferation of internet and its easy access on myriad devices, has resulted in huge amount of data generation. Initially, such data was considered disconnected and not related. But with the advent of semantic web, data has been found to be highly co-related and relevant. Organizing such huge amount of data and subsequently processing requires parallel processing framework that is both distributed and scalable. Graphical processing units (GPUs) are being actively probed in the domain of Big Data analysis, machine learning, and augmented reality since such applications are characterized by massive data spanned and generated over distributed network. GPUs provide a parallel programming framework using CUDA (Compute Unified Device Architecture) that can be utilized to efficiently collate and make inferences on these massive data-sets. Further, GPU multicores are available at commodity rates thus providing an option for cheap and low-power alternatives. The exponential growth of semantic web and the resultant generation of large-scale RDF (Resource Description Framework) triples pose new challenges in the domain of RDF-storage and retrieval. RDF data consist of triples which need to be efficiently
indexed. Following are some of the many challenges related to efficient indexing of RDF triples: • As RDF-triples extensively contain recursive redundancies, self-joins so formed are inefficient. • Self-joins also lead to large scale null values. This paper presents the research initiatives of conducting literature survey of contemporary indices including those under active research, which matches the goals as outlined in above sections. Comparing the efficiency of different variety of indices that have been suggested for large data-sets (Map reduce, B+ tree Hashed Index, 3-level-cascade hash index, braided B+ tree index, etc.)

References

- M. K. Nguyen, C. Basca and A. Bernstein, Speeding up on-disk RDF index lookups using B+Hash Trees, 2012, IOS Press, Zurich, Switzerland
- C. Weiss, P. Karras and A. Bernstein Hexastore:Sextuple Indexing for Semantic Web Data Management, 2008, VLDB Endowment, Auckland, New Zealand
- A. Matono, A Storing Scheme and A Merge Join Algorithm for RDF Query Processing, Jaban
- P. Bakkum and K. Skadron, Accelerating SQL Database Operations on a GPU with CUDA. 2010, Pittsburg, PA, USA
- M. A. Bornea, J. Dolby, A. Kementsietsidis, K. Srinivas, P. Dantressangle, O. Udrea, B. Bhattacharjee, Building an Efficient RDF Store Over a Relational Database. 2013, New York, USA.
- R. Angles and C. Gutierrez, Querying RDF Data from a Graph Database Perspective.
- M. A. Martínez-Prieto, Mario Arias, and Javier D. Fernández. Exchange and Consumption of Huge RDF Data
- Y. Luo, F. Picalausa, G. H. L. Fletcher, J. Hidders, and S. Vansummeren, Storing and Indexing Massive RDF Data Sets
- R. Blanco, P. Mika, and S. Vigna, Effective and Efficient Entity Search in RDF data
- A. MATONO, T. AMAGASA, M. YOSHIKAWA, and S. UEMURA, An Indexing Scheme for RDF and RDF Schema based on Suffix Arrays
- T. Tran and G. Ladwig, Structure Index for RDF Data. Workshop on Semantic Data Management (SemData@VLDB) 2010,September 17,2010, Singapore.
- YounHee Kim; ByungGon Kim; HaeChull Lim, ”The index organizations for
Index Terms

Computer Science

Algorithms

Keywords

RDF  Semantic Web  Graphical Processing Unit (GPU)  Compute Unified Device Architecture (CUDA)
Hashed B+ tree indexing.