Investigation Analysis On Data Prefetching And Mapreduce Techniques For User Query Processing

S.Tamil Selvan, K.A.Dhamotharan, G.Saravanan, S.Bharathidasan

tract: Big data is a group of data used for examining and extracting the useful information in recent days. Big data analytics is the application of nced analytic methods against large data comprising structured, semi-structured and unstructured data. The main aim is to store, access and ipulate the data at one place. MapReduce (MR) is the principle of Big Data Processing model with distributed large datasets. Prefetching is a ble technique for transferring the memory objects in memory hierarchy before required by processor. Many researchers carried out their research ne mapreduce method for responding to the user requested task through removing repeated tasks. But, the job completion time was not reduced g existing techniques as data prefetching was not carried out in existing methods. In order to address these problems, the existing prefetching and reduce techniques are reviewed and drawbacks of techniques are listed in this paper.

x Terms: Big data, Job completion, MapReduce, Memory hierarchy, Prefetching, Unstructured data, User Query Processing.

INTRODUCTION

data is the high-volume, high-velocity and high-variety rmation that demand cost-effective for decision making. data computing has large impact in recent years as pReduce and cloud computing methods are widespread. pReduce is the fundamental infrastructure of service when blic cloud experienced vulnerability problem. Prefetches are rted through explicit fetch operation within the program or ough logic that monitors the processor referencing pattern infer prefetching opportunities. This paper is organized as ows: Section 2 explains the review on different prefetching d mapreduce techniques for user query processing, Section explains the study and analysis of existing prefetching and preduce techniques. In section 4, possible comparison of sting methods is made. Section 5 gives the discussion and itations of the existing prefetching and mapreduce hniques for user query processing are discussed with future ection and Section 6 concludes the paper

LITERATURE SURVEY

prefetching service based task scheduler termed High rformance Scheduling Optimizer (HPSO) was designed in with prefetching to enhance the data locality for apReduce jobs. The designed method minimized map tasks using the remote data delay and enhanced Hadoop clusters. ut, job completion time was not reduced using HPSO. A erkle tree-based verification method (MtMR) was introduced [2] to guarantee high result integrity of MapReduce jobs. tMR sampled the small portion of task input/output records the private cloud and performed Merkle tree-based rification on all task input/output records.

But, the prefetching was not carried out at earlier stage to minimize the job completion time. A scalable pipeline of components constructed on the Spark engine for large-scale data processing in [3]. The main aim was to collect the data from dataset access logs for organizing them into weekly snapshots and predictive techniques to forecast the datasets. But, the latency was not minimized because prefetching was not carried out. An envisioned future large-scale computing architecture were adapted in [4] for batch processing of big data application in MapReduce model. But, computational complexity was not minimized by future large-scale computing architectures. The big data processing framework was designed in [5] to join the climate and health data and to find the correlation between the climate parameters. But, prefetching was not performed in big data processing framework. A novel intermediate data partition scheme was designed in [6] to reduce the network traffic cost for MapReduce job. Every aggregator reduced the merged traffic from multiple map tasks through addressing the aggregator placement issue. However, the data partitioning was not carried out efficiently through intermediate data partition scheme.High-Level MapReduce Query Languages was built in [7] on MR that converted the queries into an executable native MR jobs. But, the complexity was not minimized through High-Level MapReduce Query Languages. A new A* algorithm introduced in [8] reduced the Map and Reduce tasks for running the path computation on Hadoop MapReduce framework. The designed framework enhanced the feasibility and reliability. A* algorithm minimized computation time. However, MapReduce tasks was not performed by A* algorithm. A novel approach was designed in [9] to improve the metadata management performance for Hadoop in multitenant environment based on the prefetching mechanism. However, the map reduce function was not employed in the multitenant environment. A new efficient pattern mining algorithm was introduced in [10] by using MapReduce framework and Hadoop open-source implementation in the big data. A maximal AprioriMR algorithm was designed for mining condensed frequent patterns. But, the execution time was not minimized using efficient pattern mining algorithm. Secured Map Reduce (SMR) Layer was designed in [11] between the HDFS and MR Layer for improving the security and privacy. The designed model provided the privacy and security through

amil Selvan is currently working as Assistant Professor in Computer ence and Engineering in Erode Sengunthar Engineering College, de, India,E-mail: stamilselvan@esec.ac.in
Dhamotharan is currently working as Assistant Professor in mputer Science and Engineering in Erode Sengunthar Engineering lege, Erode, India, E-mail: dhamuerode@gmail.com
Saravanan is currently working as Assistant Professor in Computer ence and Engineering in Erode Sengunthar Engineering College, de, India, E-mail: gsaravanan.esec@gmail.com
harathidasan is currently working as Associate Professor in ctronics and Communication Engineering in Sree Sakthi Engineering lege, Coimbatore, India, E-mail: Ktvrbharathi@gmail.com