A Fast Distributed Non-Negative Matrix Factorization Algorithm Based on DSGD

Yan Gao, Central South University, Changsha, China
Lingjun Zhou, Central South University, Changsha, China
Baifan Chen, Central South University, Changsha, China
Xiaobing Xing, Central South University, Changsha, China

ABSTRACT

In numerous solutions, the distributed stochastic gradient descent algorithm is one of the most popular algorithms for parallelization matrix decomposition. However, in parallel calculation, the computing speed of each computing node was greatly different because of the imbalance of the computing nodes. This article reduced the data skew for all computing nodes during distributed execution to solve the problem of locking waiting. The improved algorithm on DSGD was named D-DSGD, which reduced the time consumption of the algorithm and improved the utilization rate of the nodes. Meanwhile, the dynamic step size adjusting strategy was applied to improve the convergence rate of the algorithm. To ensure the non-negative matrix decomposition, non-negative control was added into D-DSGD and the improved algorithm was named D-NMF. Compared with the existing methods, the proposed algorithm in this article has a marked impact on reducing the latency and speed of convergence.

KEYWORDS

Distributed Computing, Non-Negative Matrix Factorization, Recommender System, Scheduling Strategy, The Stochastic Gradient Descent Algorithm (SGD)

INTRODUCTION

It is well known that matrix decomposition is the critical technique in big data processing (Liu, Liao, Tang, Tang, & Liu, 2016; Zhu, Jing, & Yu. 2011). Not only that, it is widely used in many fields, such as image analysis (Bai, Li, & Hui, 2014; Liu, Yang, Guan, & Yi, 2016), biological and chemical engineering, machine learning, and data mining (Bottou, 2010; Dror, Koenigstein, Koren, & Weimer, 2011; Yin, Gao, & Zhang, 2014). It shows prominent role in the online recommendation system especially. For example, in the Netflix movie scoring competition, a variety of matrix decomposition algorithm models achieved significant performance (Ballard, Druinsky, Knight, & Schwartz, 2016). As shown in Figure 1, matrix decomposition is to simulate the original scoring matrix with the product of two low-dimensional implicit variables, namely, the user matrix W and the project matrix H. User matrix and project matrix represent the implicit variable characteristics of users and projects. On obtaining the user and project features, we can get the predicted values for

DOI: 10.4018/IJDST.2018070102
the unobserved scoring items and provide recommended results which are based on the predicted values (Desarkar, Saxena, & Sarkar, 2012).

As data size and data noise become more and more large, it becomes more difficult to extract effective and accurate information from large-scale data. Large scale data cannot be processed by the traditional single-machine matrix decomposition algorithm but completed by efficient parallel distributed algorithm. Thus, it is of great importance to study the distributed parallelization of traditional matrix decomposition algorithm (Hsieh, ChoJui, Dhillon, & Inderjit, 2011).

In recent years, researchers have proposed several parallel matrix decomposition model algorithms. These algorithms can be divided into two main categories: algorithm based on ALS (Alternating further Squares) and algorithm based on the SGD (Stochastic Gradient Descent). ALS adopts the strategy of crossover study which is a natural parallel solution (Yu, Hsieh, Si, & Dhillon, 2012). While the matrix of each column is required in the process of solving the inverse of a matrix, and different columns require diverse inverse of matrix. That reduces the efficiency of the algorithm. Due to the efficiency and easy-to-implement of SGD, SGD becomes one of the most popular algorithms in matrix decomposition with DSGD be one of the most famous algorithms (Gemulla, Nijkamp, Haas, & Sismanis, 2011). SGD is a sequential approach that cannot be executed naturally in parallel. DSGD divides the scoring matrix into several subblocks which can be divided into several subsets with the same number of subblocks. The subblocks in each subset are independent with each other and do not share any implicit variables, so they can be processed in parallel. Although DSGD realized parallelization, the following three defects remained: Firstly, the node lock waiting phenomenon of DSGD results in an increase in the overall operation time of the algorithm. Since the execution time of each chunk is different because of the uncertain data amount of the partitioned data block, the fast-executed node needs to wait for the end of other slow executed nodes. Secondly, excessive synchronization and scheduling have been used in the process of iterative operation to avoid conflict, resulting in great consumption of distributed resources. How to reduce the computing node data interaction in distributed environment is more significant. Finally, DSGD is still designed according to the idea of single-machine algorithm, its scalability is limited and cannot be of high accuracy and efficiency in mainstream distributed frameworks such as Spark.

To solve the above challenges, a distributed non-negative matrix decomposition algorithm (D-NMF) is newly proposed based on DSGD algorithm in this paper. Aiming at the problem of
Cost Efficient Implementation of Multistage Symmetric Repackable Networks
www.igi-global.com/article/cost-efficient-implementation-multistage-symmetric/60251?camid=4v1a