Procedia Computer Science

Volume 80, 2016, Pages 86–97

ICCS 2016. The International Conference on Computational Science

High Performance LDA through Collective Model Communication Optimization

Bingjing Zhang[1], Bo Peng[1,2], and Judy Qiu[1]

[1] Indiana University, Bloomington, Indiana, U.S.A.
{zhangbj,pengb,xqiu}@indiana.edu
[2] Peking University, Beijing, China

Abstract

LDA is a widely used machine learning technique for big data analysis. The application includes an inference algorithm that iteratively updates a model until it converges. A major challenge is the scaling issue in parallelization owing to the fact that the model size is huge and parallel workers need to communicate the model continually. We identify three important features of the model in parallel LDA computation: 1. The volume of model parameters required for local computation is high; 2. The time complexity of local computation is proportional to the required model size; 3. The model size shrinks as it converges. By investigating collective and asynchronous methods for model communication in different tools, we discover that optimized collective communication can improve the model update speed, thus allowing the model to converge faster. The performance improvement derives not only from accelerated communication but also from reduced iteration computation time as the model size shrinks during the model convergence. To foster faster model convergence, we design new collective communication abstractions and implement two Harp-LDA applications, "lgs" and "rtt". We compare our new approach with Yahoo! LDA and Petuum LDA, two leading implementations favoring asynchronous communication methods in the field, on a 100-node, 4000-thread Intel Haswell cluster. The experiments show that "lgs" can reach higher model likelihood with shorter or similar execution time compared with Yahoo! LDA, while "rtt" can run up to 3.9 times faster compared with Petuum LDA when achieving similar model likelihood.

Keywords: Latent Dirichlet Allocation, Parallel Algorithm, Big Model, Communication Model, Communication Optimization

1 Introduction

Latent Dirichlet Allocation (LDA) [1] is an important machine learning technique that has been widely used in areas such as text mining, advertising, recommender systems, network analysis, and genetics. Though extensive research on this topic exists, the data analysis community is

Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2016

doi:10.1016/j.procs.2016.05.300

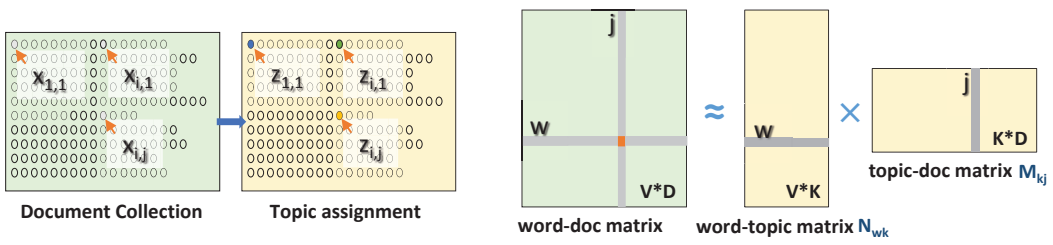

Figure 1: (a) Topics Discovery (b) View of Matrix Decomposition

still endeavoring to scale it to web-scale corpora to explore more subtle semantics with a big model which may contain billions of model parameters [5].

We identify that the size of the model required for local computation is so large that sending such data to every worker results in communication bottlenecks. The required computation time is great due to the large model size. In addition, the model size shrinks as the model converges. As a result, a faster communication method can speed up the model convergence, in which the model size shrinks and reduces the iteration execution time.

By guaranteeing the algorithm correctness, various model communication strategies are developed in parallel LDA. Existing solutions favor asynchronous communication methods, since it not only avoids global waiting but also quickly makes the model update visible to other workers and thereby boosts model convergence. We propose a more efficient approach in which the model communication speed is improved upon with optimized collective communication methods. We abstract three new communication operations and implement them on top of Harp [15]. We develop two Harp-LDA applications and compare them with Yahoo! LDA[1] and Petuum LDA[2], two well-known implementations in the domain. This is done on three datasets each with a total of 10 billion model parameters. The results on a 100-node, 4000-thread Intel Haswell cluster show that collective communication optimizations can significantly reduce communication overhead and improve model convergence speed.

The following sections describe: the background of LDA application (Section 2), the big model problem in parallel LDA (Section 3), the model communication challenges in parallel LDA and related work (Section 4), Harp-LDA implementations (Section 5), experiments and performance comparisons (Section 6), and the conclusion (Section 7).

2 Background

LDA modeling techniques can find latent structures inside the training data which are abstracted as a collection of documents, each with a bag of words. It models each document as a mixture of latent topics and each topic as a multinomial distribution over words. Then an inference algorithm works iteratively until it outputs the converged topic assignments for the training data (see Figure 1(a)). Similar to Singular Value Decomposition (see Figure 1(b)), LDA can be viewed as a sparse matrix decomposition technique on a word-document matrix, but it roots on a probabilistic foundation and has different computation characteristics.

[1] https://github.com/sudar/Yahoo_LDA
[2] https://github.com/petuum/bosen/wiki/Latent-Dirichlet-Allocation