Associate Professorship of Coding and Cryptography
TUM Department of Electrical and Computer Engineering
Technical University of Munich

Forschungspraxis

Adaptive compute strategies for distributed learning

Associate Professorship of Coding and Cryptography
TUM Department of Electrical and Computer Engineering
Technical University of Munich

We consider the master/server computing model. A master possesses a large amount of data and wants to run a machine learning algorithm on this data. The master offloads the data to several worker machines to parallelize the computation and reduce the time it takes. If the master waits for all the workers, he is limited by the slowest worker. This is known as the straggler problem.

To mitigate the straggler problem, the master can distribute the data with redundancy to the workers. However, adding redundancy to the workers increases the computing time. On the other hand, stochastic gradient descent is known to converge even if the master does not wait for all the workers per iteration. Therefore, a tradeoff between redundancy, and convergence speed arises.

It is shown that for convex loss functions, the master can assign data without redundancy to the workers and chooses the number of workers to wait for per iteration to minimize the waiting time.

The goal of this project is to have the best of both worlds. More precisely, we want to design coding techniques that potentially change the redundancy of the data in an adaptive manner. In addition, the technique allows the master to adaptively choose the number of workers it waits for at every iteration. Both assignment strategy and waiting strategy aim to minimize the time spent on running the machine learning algorithm.

The coding technique we aim to introduce is tailored to the problem at hand. If communication between the master and workers is fast, e.g., serverless compute service, then changing the redundancy is affordable. Otherwise, in settings such as internet of things and edge computing, the master may want to fix the redundancy a priori to a designed value and choose to adaptively wait for a different number of workers per iterations.

It is expected to program a distributed stochastic gradient descent on Google cloud platform, or Amazon Web Services to validate and/or design our system's parameters.

Prerequisites

Knowledge of coding theory, linear algebra and machine learning

Good programming skills

Self-motivated and eager to learn

Bonus: knowledge of probability theory and convex optimization

Contact

rawad.bitar@tum.de

https://sites.google.com/site/rawadbitar1

Advisors

Rawad Bitar