Spark: Fast Cluster Computing

Evaluation of Hyper-parameters Techniques

Open Source Project Presentation

Ridwan Muheeb
rm2084
Introduction

• Apache Spark is a framework for large-scale data processing that distribute large data jobs across CPU clusters.

• A programming model that offers significant reuse of intermediate results/datasets.

• Use the concept of Resilient Distributed Dataset (RDD) abstraction to store intermediates of cluster computations and Lineage (log of transformations on RDD) for fault-tolerance and locality-aware scheduling.

• Offers iterative machine learning and graph jobs processing by interactively loading large dataset into aggregate memory cluster and then perform multiple ad-hoc queries.
ML in Spark

• In Spark, MLib is the defacto machine learning library that provides a high-level API built on-top of DataFrames which supports ML workflow and specification of their parameters.

• To improve the models predictive power and reduce training time, hyperparameters are added to the model prior.

• Hyperparameters (HP) tuning is essential because it involves basically the process of optimizing machine learning configurations to have the best performance possible out of a model.

• They can be tuned systematically or domain experts can provide feedback.
Aim/Goal

• The goal of this project is to evaluate the performance of hyperparameters tuning strategies available in Spark for largescale and distributed workloads.

• The project aim to measure performance of Hyperopt, a library for ML hyperparameter tuning in Python, and Apache Spark Mlib.

• The output of the project is to help researchers decide best tuning strategy in Spark.
Journey so far...

• Not much.

• So far, I have been able to:
  • Identify the Spark frameworks needed to achieve the aim of the project.
  • Identify some datasets to evaluate the hyperparameters tuning techniques on.

• Further work will involve:
  • Identify appropriate machine learning models to test the HP techniques.
  • Coding the models and testing the HP techniques.
  • Critical HP evaluation using the identified distributed machine learning workloads.
  • Report writing, proofreading and submission.
Bibliography

• Matei, Zaharia et al. (2012). Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing. NSDI.
