Large Scale Distributed Data Science from Scratch using Apache Spark 2.0

Dr. James G. Shanahan
Church and Duncan Group
University of California, Berkeley
James.Shanahan@gmail.com

Liang Dai
University of California Santa Cruz
liangdai16@gmail.com

ABSTRACT

Apache Spark is an open-source cluster computing framework. It has emerged as the next generation big data processing engine, overtaking Hadoop MapReduce which helped ignite the big data revolution. Spark maintains MapReduce's linear scalability and fault tolerance, but extends it in a few important ways: it is much faster (100 times faster for certain applications), much easier to program in due to its rich APIs in Python, Java, Scala, SQL and R (MapReduce has 2 core calls) , and its core data abstraction, the distributed data frame. In addition, it goes far beyond batch applications to support a variety of compute-intensive tasks, including interactive queries, streaming, machine learning, and graph processing.

With massive amounts of computational power, deep learning has been shown to produce state-of-the-art results on various tasks in different fields like computer vision, automatic speech recognition, natural language processing and online advertising targeting. Thanks to the open-source frameworks, e.g. Torch, Theano, Caffe, MxNet, Keras and TensorFlow, we can build deep learning model in a much easier way. Among all these framework, TensorFlow is probably the most popular open source deep learning library. TensorFlow 1.0 was released recently, which provide a more stable, flexible and powerful computation tool for numerical computation using data flow graphs. Keras is a high-level neural networks library, written in Python and capable of running on top of either TensorFlow or Theano. It was developed with a focus on enabling fast experimentation.

This tutorial will provide an accessible introduction to large-scale distributed machine learning and data mining, and to Spark and its potential to revolutionize academic and commercial data science practices. It is divided into three parts: the first part will cover fundamental Spark concepts, including Spark Core, functional programming ala map-reduce, data frames, the Spark Shell, Spark Streaming, Spark SQL, MLlib, and more; the second part will focus on hands-on algorithmic design and development with Spark (developing algorithms from scratch such as decision tree learning, association rule mining (aPriori), graph processing algorithms such as pagerank/shortest path, gradient descent algorithms such as support vectors machines and matrix factorization. Industrial applications and deployments of Spark will also be presented.; the third part will introduce deep learning concepts, how to implement a deep learning model through TensorFlow, Keras and run the model on Spark. Example code will be made available in python (pySpark) notebooks.

WWW 2017 Companion, April 3-7, 2017, Perth, Australia.
ACM 978-1-4503-4914-7/17/04.
http://dx.doi.org/10.1145/3041021.3051108

Categories and Subject Descriptors

Distributed systems, machine learning, Tutorial

Keywords

Distributed systems, HDFS, Spark, Hadoop, Deep learning, large scale machine learning, mobile advertising

1. INTRODUCTION

This tutorial is for data scientists who may not already be familiar with Spark, distributed systems, and large machine learning. This tutorial introduces the underlying statistical and algorithmic principles required to develop scalable machine learning pipelines, and provides hands-on experience using PySpark, TensorFlow and Keras. It presents an integrated view of data processing by highlighting the various components of Spark pipelines, including exploratory data analysis, feature extraction, supervised learning, and model evaluation. Students will use Spark to implement scalable algorithms for fundamental statistical models. Data intensive industrial applications and deployments of Spark will also be presented, in fields such as mobile advertising.

We present an integrated view of data processing by highlighting the various components of these pipelines, including exploratory data analysis, feature extraction, supervised learning, and model evaluation. You will gain hands-on experience applying these principles using Apache Spark, a cluster computing system well suited for large-scale machine learning tasks. You will implement scalable algorithms from fundamental statistical models (linear regression, logistic regression, matrix factorization, principal component analysis) to deep learning model while tackling key problems from various domains: mobile advertising, personalized recommendation, and consumer segmentation.

The emphasis of this tutorial is scalability and the tradeoffs associated with distributed processing of large datasets. The tutorial will cover "core" data science topics (e.g., gradient descent) as well as related topics in the broader area of human language technologies (e.g., distributed parameter estimation, graphs algorithms). Content will include general discussions of algorithm design, presentation of illustrative algorithms, relevant case studies, as well as practical advice in writing Spark programs and running Spark clusters.

Participants will deploy Spark on their multicore laptops and run and develop examples there. In addition, we plan to work with Amazon Web Services (AWS) and get participants in this tutorial (free) access to Amazon's Elastic Compute Cloud (EC2). With this "utility computing" service, participants will be able to rapidly provision Spark clusters on the fly without needing to purchase any hardware.

Target audience: The tutorial is targeted to most WWW attendees, both industry practitioners and researchers who wish to learn best practices of large scale data science using next

generation tools. The level of the tutorial can be considered introductory with hands-on exposure to algorithmic development (and pySpark the python API to Spark) and deep learning in Spark.

Prerequisite knowledge of audience: Programming background; comfort with mathematical and algorithmic reasoning; familiarity with basic machine learning concepts; exposure to algorithms, probability, linear algebra and calculus; experience with Python (or the ability to learn it quickly). All exercises will use PySpark, but previous experience with Spark or distributed computing is NOT required.

2. OUTLINES

Chapter	Topics
Spark Introduction and Hello World	History of Spark Introduction to data analysis with Spark Downloading Spark and getting started on your local machine
Parallel computing	Divide and conquer, Semaphores, Barriers, Shared nothing architectures
Core Spark	Spark intro and basics Functional programming Transformations and actions, Map-Reduce patterns, Dataframes, RDD (no keys), pySpark, Pair RDDs, Scala, Spark Shell, Broadcast variables
Spark APIs	Java, Scala, Python, R, SQL
Data analysis and handling with Spark	Tools for exploratory data analysis, Standardization, Reservoir Sampling, SparkSQL; Join, Statistics in SPARK;
Algorithms and programming in Spark	Algorithmic design and development with Spark Developing algorithms from scratch • Decision tree learning • Naïve Bayes Association rule mining • aPriori algorithm Graph processing algorithms • Pagerank • Shortest path • Friend of friends • TextRank Unsupervised algorithms • Expectation maximization Gradient descent algorithms • support vectors machines matrix factorization
Spark at Scale	Install Spark on your muiltcore laptop Run Spark on an EC2 cluster
Spark libraries	SparkSQL, MLlib, GraphX, Spark Streaming, Spark deployments
Spark deployments and case studies	Mobile advertising, Recommendation engines
Deep learning	Deep learning concept, TensorFlow, Keras, Run Keras using TensorFlow backend in Spark

3. PRESENTER BIOGRAPHY

Dr. James G. Shanahan, CEO and Chief Scientist at Church and Duncan Group and UC Berkeley

Dr. James G. Shanahan has spent the past 25 years developing and researching cutting-edge artificial intelligent systems. He has (co) founded several companies including: Church and Duncan Group Inc. (2007), a boutique consultancy in large scale AI which he runs in San Francisco; RTBFast (2012), a real-time bidding engine infrastructure play for digital advertising systems; and Document Souls (1999), a document-centric anticipatory information system. In 2012 he went in-house as the SVP of Data Science and Chief Scientist at NativeX, a mobile ad network that got acquired by MobVista in early 2016. In addition, he has held appointments at AT&T (Executive Director of Research), Turn Inc. (founding chief scientist), Xerox Research, Mitsubishi Research, and at Clairvoyance Corp (a spinoff research lab from CMU).

Dr. Shanahan has been affiliated with the University of California at Berkeley (and Santa Cruz) since 2008 where he teaches graduate courses on big data analytics, machine learning, and stochastic optimization. He also advises several high-tech startups (including Quixey, Aylien, VoxEdu, and others) and is executive VP of science and technology at Irish Innovation Center (IIC). He has published six books, more than 50 research publications, and over 20 patents in the areas of machine learning and information processing. Dr. Shanahan received his PhD in engineering mathematics from the University of Bristol, U. K., and holds a Bachelor of Science degree from the University of Limerick, Ireland. He is a EU Marie Curie fellow. In 2011 he was selected as a member of the Silicon Valley 50 (Top 50 Irish Americans in Technology).

Liang Dai, UC Santa Cruz and Applied Research Scientist, Facebook

Liang Dai is a Ph.D. candidate in Technology Information and Management department, UC Santa Cruz. There he does research in data mining on digital marketing, including campaign evaluation, online experiment design, customer value improvement, etc. Liang received the B.S. and the M.S. from Information Science and Electronic Engineering department, Zhejiang University, China. Liang is also working as a applied research scientist in Facebook, focusing on data modeling for ads product. He has hands-on experience on end to end large scale data mining projects in distributed platform, e.g. AWS, Hadoop, Spark, etc.

4. REFERENCES

[1] Holden Karau, Andy Konwinski, Patrick Wendell, Matei Zahari, Learning Spark: Lightning-Fast Big Data Analysis, O'Reilly Media, January 2015

[2] Advanced Analytics with Apache Spark: The Book, Sean Owen, Sandy Ryza, Uri Laserson,

Josh Wills, O'Reilly Media, April 2015

[3] Matei Zaharia. Spark: In-Memory Cluster Computing for Iterative and Interactive

Applications. Invited Talk at NIPS 2011 Big Learning Workshop: Algorithms, Systems, and Tools for Learning at Scale.

[4] "Cluster Mode Overview - Spark 1.2.0 Documentation - Cluster Manager Types". apache.org. Apache Foundation. 2014-12-18. Retrieved2015-01-18.

[5] https://www.gitbook.com/book/databricks/databricks-spark-reference-applications/details

[6] Shanahan, J. G., Kurra, G. Web Advertising: Business Models, Technologies and Issues in Information Retrieval, Edited by Massimo Melucci and Ricardo Baeza-Yates, 2010

[7] Biswanath Panda, Joshua S. Herbach, Sugato Basu, and Roberto J. Bayardo. 2009. PLANET: massively parallel learning of tree ensembles with MapReduce. Proc. VLDB Endow. 2, 2 (August 2009), 1426-1437. DOI=10.14778/1687553.1687569 http://dx.doi.org/10.14778/1687553.1687569

[8] Gábor Takács et al (2008). Matrix factorization and neighbor based algorithms for the Netflix prize problem. In: Proceedings of the 2008 ACM Conference on Recommender Systems, Lausanne, Switzerland, October 23 - 25, 267-274.

[9] Patrick Ott (2008). Incremental Matrix Factorization for Collaborative Filtering. Science, Technology and Design 01/2008, Anhalt University of Applied Sciences.

[10] TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[11] François Chollet, Keras, (2015), GitHub repository: https://github.com/fchollet/keras

[12] Joeri Hermans and CERN IT-DB, Distributed Keras : Distributed Deep Learning with Apache Spark and Keras, (2016), GitHub repository: https://github.com/JoeriHermans/dist-keras/