Machine Learning for Big Data

Tyson Condie[1] Paul Mineiro[1] Neoklis Polyzotis[2] Markus Weimer[1]

[1] Cloud and Information Services Lab, Microsoft
One Microsoft Way, Redmond, USA
{tcondie, pmineiro, mweimer}@microsoft.com

[2] University of California, Santa Cruz
1156 High Street, Santa Cruz. California, USA
alkis@ucsc.edu

ABSTRACT
Statistical Machine Learning has undergone a phase transition from a pure academic endeavor to being one of the main drivers of modern commerce and science. Even more so, recent results such as those on tera-scale learning [1] and on very large neural networks [2] suggest that scale is an important ingredient in quality modeling. This tutorial introduces current applications, techniques and systems with the aim of cross-fertilizing research between the database and machine learning communities.

The tutorial covers current large scale applications of Machine Learning, their computational model and the workflow behind building those. Based on this foundation, we present the current state-of-the-art in systems support in the bulk of the tutorial. We also identify critical gaps in the state-of-the-art. This leads to the closing of the seminar, where we introduce two sets of open research questions: Better systems support for the already established use cases of Machine Learning and support for recent advances in Machine Learning research.

Categories and Subject Descriptors
H.2.8 [**Database Applications**]: Data mining.

General Terms
Algorithms, Performance, Languages, Theory.

Keywords
Machine learning, big data, databases.

1. INTRODUCTION
The democratization of data has fueled the collection of (even more!) massive data sets. Platforms for large-scale analytics are being built for extracting insights that enhance services and optimize operations. However, the current suite of platforms can only capture a small fraction of machine learning algorithms at scale. Those that fall out of scope induce solutions that abuse the intended programming model or motivate the implementation of a separate system. This leads to production pipelines built out of bailing wire and glue code for moving data in-and-out of different subsystems. The problem is further complicated by the assumption that each system assumes ownership of the entire machine, or at best the VM, forcing administrators to divide the cluster into machine subsets; assigning a single system to each. Resource negotiators like YARN [3] and Mesos [4] are being built to address this problem with a thin virtualization layer that multiplexes higher-level systems on a single unified machine cluster. Yet, the data scientist is still left with the problem of finding the right system and programming model for their data-analysis task, assuming one even exists, and staging their solution in production.

We believe that the database community can provide the appropriate data management toolkits for the data scientist to operate at scale. Large-scale (distributed) machine learning techniques involve many of the same issues parallel databases faced decades ago; specifically, efficient shared-nothing architectures that support a relational-algebra programming model. Machine-learning algorithms also have characteristics (most notably recursion) that fall directly in the scope of deductive database query processing techniques. And finally, we see a longer term need for real-time learning support, which will leverage work from streaming database systems.

This seminar aims to inform the database community about typical use cases and workflows in the machine-learning domain. One goal in particular is to highlight the differences to traditional data processing and thus to outline potential research topics. Examples include:

- The need for iteration: Machine Learning algorithms almost exclusively are iterative.
- Non-standard fault tolerance policies: Some machine learning algorithms can deal with unavailable partitions.
- Aggregation functions over very large objects: many machine-learning algorithms involve the aggregation of objects whose size is in the same order of magnitude as the available main memory.

We hope that this seminar adds to the ongoing alignment of machine learning and database research.

2. TARGET AUDIENCE
The seminar is targeted at database-systems practitioners and researchers who are interested in the application of machine learning in Big Data analytics as well as research on systems that support large scale machine learning. The seminar does not require prior familiarity with machine learning techniques.

3. SEMINAR OUTLINE
The seminar is structured into three sections, moving from machine learning practice and theory, to current systems support for machine learning at scale, to a research agenda for the database community.

SIGMOD'13, June 22–27, 2013, New York, New York, USA.

3.1 Machine Learning Applications and Practice

In this part of the seminar, we present a quick overview over the theory and, more importantly, practice of machine learning. The goal of this section is to give the audience a grasp of the typical uses of and challenges faced by machine learning in practice.

Drawing from the industrial expertise of the presenters in large and small machine learning deployments, we develop a prototypical workflow of machine learning projects. This workflow consists of three phases: Example Formation, Modeling and Deployment. We ground the modeling phase in the machine learning literature, and introduce the Statistical Query Model [5] that underlies many machine learning algorithms and their scalable implementations [1] [6]. We also discuss the data plumbing that goes into a typical machine learning project as well as part of the feature extraction and the deployment phases of the workflow. We present this workflow using example applications familiar to the audience such as email spam filtering and server-hardware fault prediction.

3.2 Systems Support for Large-Scale Machine Learning

In this section, we discuss the various tools and systems used by machine learning practitioners today in more detail. We cover this material according to the prototypical workflow introduced above: (a) Example Formation, (b) Modeling and learning and (c) Deployment.

For Example Formation, we cover the use of Pig [7], MapReduce [8], databases and scripting.

The tools for Modeling and learning are discussed based on their computational model. We start with relational algebra inspired systems: MapReduce and its use in Mahout [9] and Spark [10]. We then introduce approaches that are grounded in numerical computation using VW [1] as the example. Lastly, we discuss graph processing inspired approaches: Pregel [11] and its Open Source implementation Giraph [12] and GraphLab [13].

Deployment is very application-specific, but we cover examples from the domains of recommender systems and spam-filters for email, in addition to the running example of machine fault prediction.

We close this section with a discussion of some of the gaps in current systems with respect to their support for large-scale machine learning.

3.3 Open Research Issues

In this last section, we outline open problems for large-scale machine learning and how they relate to database research. This section is split into two aspects:

Better systems support for the approaches covered in this seminar: Those open problems include: (1) Query and runtime optimizers for iterative, distributed programs. (2) Declarative specifications and optimizers for asynchronous computations and (3) Semantic models of non-standard fault tolerance policies. While each of these has been studied in the past, the specifics of machine learning computation and the latter's importance in Big Data analysis make it worthwhile directing the attention of the database community towards them.

Systems support for recent results in machine-learning research: Approaches such as deep belief networks [14] [15] and graphical models have recently received lots of attention in the machine learning community [2] and even the mainstream media [16]. However, their systems support is currently severely lacking due to their high communication load. We give an overview of those challenges and the current state of the art solutions, as well as their shortcomings.

It is our hope that this section fosters the convergence of machine learning and databases inside big data systems.

4. REFERENCES

[1] A. Agarwal, O. Chapelle, M. Dudik and J. Langford, "A Reliable Effective Terascale Linear Learning System," arXiv.org, 2012.

[2] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, Q. Le, M. Mao, A. Senior, P. Tucker, K. Yang and A. Ng, "Large Scale Distributed Deep Networks," in *Advances in Neural Information Processing Systems*, 2013.

[3] The Apache Project, "Apache Hadoop NextGen MapReduce (YARN)," The Apache Project, [Online]. Available: http://hadoop.apache.org/docs/r0.23.0/hadoop-yarn/hadoop-yarn-site/YARN.html.

[4] B. Hindman, A. Konwinski, M. Zaharia and I. Stoica, "A Common Substrate for Cluster Computing," in *HotCloud*, 2009.

[5] M. Kearns, "Efficient noise-tolerant learning from statistical queries," *Journal of the ACM,* pp. 392-401, 1998.

[6] C.-T. Chu, S. K. Kim, Y.-A. Lin, Y. Yu, G. Bradski and A. Y. Ng, "Map-Reduce for Machine Learning on Multicore," in *Advances in Neural Information Processing Systems 19*, Cambridge, MA, 2007.

[7] The Apache Foundation, "Apache Pig," 11 12 2012. [Online]. Available: http://pig.apache.org/.

[8] J. Dean and S. Ghemat, "MapReduce: simplified data processing on large clusters," *Communications of the ACM,* vol. 51, pp. 107-113, 2008.

[9] The Apache Mahout Project, "Apache Mahout," 17 9 2012. [Online]. Available: http://mahout.apache.org/. [Accessed 17 9 2012].

[10] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker and I. Stoica, "Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing," in *USENIX NSDI*, San Jose, CA, 2012.

[11] G. Malewicz, M. H. Austern, A. J. Bik, J. C. Dehnert, I. Horn, N. Leiser and G. Czajkowski, "Pregel: a system for large-scale graph processing," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, Indianapolis, Indiana, USA, 2010.

[12] The Apache Software Foundation, "Apache Giraph," [Online]. Available: http://giraph.apache.org/.

[13] Y. Low, D. Bickson, J. Gonzalez, C. Guestrin, A. Kyrola and J. M. Hellerstein, "Distributed GraphLab: a framework for machine learning and data mining in the cloud," *Proceedings of the VLDB Endowment,* vol. 5, no. 8, pp. 716-727, April 2012.

[14] G. Hinton, "Learning Multiple Layers of Representation," *Trends in Cognitive Sciences,* vol. 11, pp. 428-434, 2007.

[15] G. E. Hinton, S. Osindero and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation,* vol. 18, pp. 1526-1554, 2006.

[16] J. Markoff, "How Many Computers to Identify a Cat? 16,000," The New York Times, 25 June 2012. [Online]. Available: http://www.nytimes.com/2012/06/26/technology/in-a-big-network-of-computers-evidence-of-machine-learning.html. [Accessed 11 December 2012].

[17] A. Smola, A. Ahmed and M. Weimer, "WWW 2012 Tutorial: New Templates for Scalable Data Analysis," June 2012. [Online]. Available: http://www2012.wwwconference.org/program/tutorials/ and http://cs.markusweimer.com/2012/04/06/www-2012-tutorial-new-templates-for-scalable-data-analysis/.

[18] G. Dror, N. Koenigstein, Y. Koren and M. Weimer, "The Yahoo! Music Dataset and KDD-Cup'11," in *Proceedings of KDDCup 2011*, San Diego, CA, 2011.