Tutorial on Benchmarking Big Data Analytics Systems

Todor Ivanov
todor@dbis.cs.uni-frankfurt.de
Frankfurt Big Data Lab, Goethe University
Frankfurt am Main, Germany

Rekha Singhal
rekha.singhal@tcs.com
TCS Research
Mumbai, India

ABSTRACT

The proliferation of big data technology and faster computing systems led to pervasions of AI based solutions in our life. There is need to understand how to benchmark systems used to build AI based solutions that have a complex pipeline of pre-processing, statistical analysis, machine learning and deep learning on data to build prediction models. Solution architects, engineers and researchers may use open-source technology or proprietary systems based on desired performance requirements. The performance metrics may be data pre-processing time, model training time and model inference time. We do not see a single benchmark answering all questions of solution architects and researchers. This tutorial covers both practical and research questions on relevant Big Data and Analytics benchmarks.

CCS CONCEPTS

• **Computer systems organization → Distributed architectures**.

KEYWORDS

Big Data, Analytics, ML, AI, Benchmarking

ACM Reference Format:
Todor Ivanov and Rekha Singhal. 2020. Tutorial on Benchmarking Big Data Analytics Systems. In *ACM/SPEC International Conference on Performance Engineering Companion (ICPE '20 Companion), April 20–24, 2020, Edmonton, AB, Canada.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3375555.3383121

1 INTRODUCTION

In the age of Big Data, often characterized by the so called 3Vs (*Volume, Velocity and Variety*) [31], it is essential to use the right tools and best practices when implementing AI applications. Traditionally, benchmarking tools and methodologies have been used to compare different technologies both in terms of performance and functionality [16]. With the growing number of open source and enterprise tools in the Big Data Ecosystem [17], the need of standardized Big Data Benchmarks that provide accurate comparison between these new technologies has become very important [5]. The use of big data technologies for building machine-learning [39] and deep-learning [50] pipelines, and models has introduced more complexity in choosing the right model building framework, libraries and hardware architecture.

ICPE '20 Companion, April 20–24, 2020, Edmonton, AB, Canada

ACM ISBN 978-1-4503-7109-4/20/04.
https://doi.org/10.1145/3375555.3383121

AI [3] has motivated the advances in hardware development, such as new hardware accelerators and configurable components [36] (e.g. NVMs (Non-Volatile Memory) [4], GPUs (Graphics Processing Unit) [42], FPGAs (Field Programmable Gate Array) [48, 49], TPUs (Tensor Processing Units) [9] and more), which suggest a complete rewriting of the existing software stack [29]. Such major changes in the backend systems impact both the processing and storage layers. In order to optimize and validate the benefits of the new software stack, suitable and standardized Big Data benchmarks comprising of machine-learning and deep-learning workloads are necessary.

Historically, technical benchmarking can be seen as a process of applying transparent and common methodologies to compare systems or software technologies. Jim Gray back in 1992 [16] described benchmarking as follows: *"This quantitative comparison starts with the definition of a benchmark or workload. The benchmark is run on several different systems, and the performance and price of each system is measured and recorded. Performance is typically a throughput metric (work/second) and price is typically a five-year cost-of-ownership metric. Together, they give a price/performance ratio."*

In short, we can summarize that a software benchmark is a program used for comparison of software products/tools executing on a pre-configured hardware environment. There are different types of benchmarks that focus on specific functionalities:

Micro-benchmarks are either a program or routine to measure and test the performance of a single component or task [38]. They are used to evaluate either individual system components or specific system behaviors (or functions of codes) [19]. Micro-benchmarks report simple and well-defined quantities such as elapsed time, rate of operations, bandwidth, or latency [38]. Typically, they are developed for a specific technology, which reduces their complexity and development overhead. Popular micro-benchmark examples also part of the Hadoop binaries are WordCount, TestDFSIO, Pi, K-means, HiveBench and many others.

Application-level benchmarks also known as **End-to-end benchmarks** are designed to evaluate the entire system using typical application scenarios, each scenario corresponds to a collection of related workloads [19]. Typically, these type of benchmarks are more complex and are implemented using multiple technologies, which makes them significantly harder to develop. For example application-level Big Data benchmarks are the one standardized by the Transaction Processing Performance Council (TPC) [47] such as TPC-H, TPC-DS, BigBench(TPCx-BB) and many others.

Benchmark suites are combinations of different micro and/or end-to-end (application-level) benchmarks and these suites aim to provide comprehensive benchmarking solutions [19]. Examples for Big Data benchmark suites are HiBench [24], SparkBench [33], CloudSuite [8], BigDataBench [22], PUMA [1] and many others.

Another important distinction between benchmarks is if they are **standardized** by an official organization (like SPEC [45] or TPC [47]) or **not standardized** (typically developed by a vendor or research organization).

2 TUTORIAL STRUCTURE

The tutorial is organized for 3 hours with half an hour break in between. The tutorial covers the research questions and available benchmarks in the big data analytics domain. It aims to answer relevant questions such as:

- What benchmark to pick for a particular application?
- How to distinguish different available benchmarks for a given application?
- How do we address the different application requirements in terms of schema, heterogeneity of data types and technologies?
- How to pick the right benchmark type (micro-benchmark, application and suites) for particular use case?

The tutorial will focus on the following aspects of the big-data-analytic-systems-benchmarks:

- Summarize the Big Data challenges, requirements and features that emerging new technologies and benchmarks should address.
- Present an extensive overview of the current benchmark initiatives and organizations developed as part of the DataBench project classified according to relevant areas, technologies and architecture stacks.
- Present in detail popular and representative Machine Learning and Big Data benchmarks.
- Outline popular tools and methodologies for evaluating technologies and platforms by utilizing existing benchmarks.

3 BIG DATA ANALYTICS TECHNOLOGIES

Due to the growing number of new data platforms like Hybrid Transaction/Analytical Processing (HTAP) ([30, 35]), Distributed Parallel Processing Engines ([41], [18], [43], [13], etc.), Big Data Management ([2]), SQL-on-Hadoop-alike ([20], [44], [23], etc.) and Analytics Systems ([21]) integrating Machine Learning ([34], [32]), Deep Learning ([46]) and more, the emerging benchmarks try to follow the trend to stress these new system features. This makes the currently standardized benchmarks (such as TPC-C, TPC-H, etc.) only partially relevant for the emerging Big Data Management systems as they offer new features that require new analytics benchmarks.

Also, the data-driven nature of machine/deep learning workloads motivated research and development in specialized hardware such as TPU, GPU, etc.. The modern benchmark for big data analytics systems shall encompass heterogeneous middle-ware and hardware architectures.

4 BIG DATA BENCHMARKS

Figure 1 is an attempt to classify and categorize the most popular Big Data and Analytics benchmarks. We divided the benchmarks in six categories according to their workload, data type and use of Big Data technologies. These categories are *mirco-benchmarks, Big Data and SQL-on-Hadoop benchmarks, streaming benchmarks, machine learning and deep learning benchmarks, graph benchmarks and new emerging benchmarks.* Below are summarized popular benchmarks that will be presented in the tutorial:

4.1 BigBench

BigBench [6, 15] is an end-to-end application-level big data benchmark that represents a data model simulating the volume, velocity and variety characteristics of a big data system, together with a synthetic data generator for structured, semi-structured and unstructured data. The structured part of the retail data model is adopted from the TPC-DS benchmark and further extended with semi-structured (registered and guest user clicks) and unstructured data (product reviews). It consists of 30 complex queries. In 2016, TPC standardized BigBench as TPCx-BB.

BigBench V2 [14] benchmark addresses some of the limitation of the BigBench (TPCx-BB) benchmark. BigBench V2 separates from TPC-DS with a simple data model. The new data model still has the variety of structured, semi-structured, and unstructured data as the original BigBench data model. The difference is that the structured part has only six tables that capture necessary information about users (customers), products, web pages, stores, online sales and store sales. BigBench V2 mandates late binding by requiring query processing to be done directly on key-value web-logs rather than a pre-parsed form of it.

BigBench Streaming Extension [26] extends the BigBench V2 benchmark with a data streaming component that simulates typical statistical and predictive analytics queries in a retail business scenario. The goal is to preserve the existing BigBench design and introduce a new streaming component that supports two data streaming modes: *active and passive.* In *active mode*, the data stream generation and processing happen in parallel, whereas in *passive mode*, the data stream is pre-generated in advance before the actual stream processing. The stream workload consists of five queries inspired by the existing 30 BigBench queries.

ADABench [40] is an end-to-end ML benchmark covering the complete ML lifecycle, from data preparation all the way to inference. It covers 16 real business use cases including different scale ranges. Currently six cases are implemented (in Python and Spark) covering various dimensions of analytics in the retail business vertical.

4.2 BigDataBench

BigDataBench [7] is an open source Big Data benchmark suite [22] consisting of 14 data sets and 33 workloads. Six of the 14 data sets are real-world based, generated using the BDGS [12] data generator. The generated data types include text, graph, and table data, and are fully scalable. The 33 workloads are divided into five common application domains: *search engine, social networks, electronic commerce, multimedia analytics, and bioinformatics.*

4.3 MLPerf

The MLPerf[1] [10, 11] project aims to build a common set of benchmarks that enables the machine learning field to measure system performance for both training and inference from mobile devices

[1] www.mlperf.org

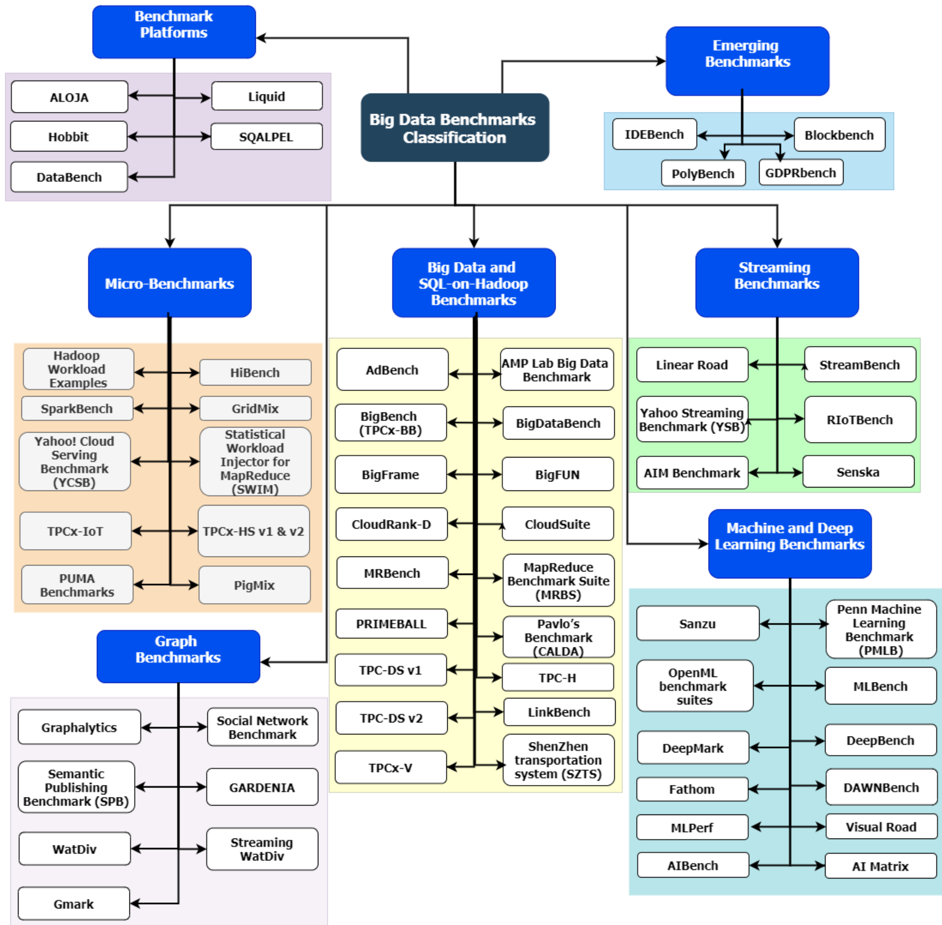

Figure 1: Big Data Benchmarks Classification [25]

to cloud services. It aims to collect publicly available data sets and models for the following problems: *Image classification, Object detection, Translation, Recommendation, Reinforcement Learning, Speech to text and Sentiment Analysis*.

4.4 DataBench

DataBench[2] [27, 37] is a three year EU-funded project that investigates existing Big Data benchmarking tools and projects, identifies the main gaps and provides a robust set of metrics to compare technical results coming from those tools. The DataBench Toolbox is a one-stop-shop for Big Data Benchmarking, offering multiple benefits for different kind of users and businesses.

4.5 ABench

ABench [28] is *Big Data Architecture Stack Benchmark* that targets the representation and comparison of different Big Data architecture patterns. The benchmark framework shall stress test the common application business requirements (e.g. retail analytics, retail operational, etc.), big data technologies functionalities and best practice implementation architectures. The benchmark framework should have an open source implementation and extendable design as well as easy to be setup and extend. It should include data generator, public data sets and existing benchmarks to simulate workloads that stress test the best practice Big Data architectures.

[2]www.databench.eu

5 CONCLUSIONS

AI is pervasive in our life and there is need to understand how to benchmark systems, which are used to build machine-learning and deep-learning based solutions using emerging big data technology stacks. This tutorial covers the research questions and available benchmarks in this domain. We have discussed in detail popular benchmarks such as BigBench, BigDataBench, MLPerf and the DataBench project.

ACKNOWLEDGMENTS

This work has been partially funded by the European Commission H2020 project DataBench - Evidence Based Big Data Benchmarking to Improve Business Performance, under project No. 780966. This work expresses the opinions of the authors and not necessarily those of the European Commission. The European Commission is not liable for any use that may be made of the information contained in this work. The authors thank all the participants in the project for discussions and common work.

REFERENCES

[1] Faraz Ahmad. 2012. PUMA Benchmarks. https://engineering.purdue.edu/~puma/pumabenchmarks.htm
[2] AsterixDB. 2020. asterixdb.apache.org.
[3] Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2019. Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* 41, 2 (2019), 423–443.
[4] Jalil Boukhobza, Stéphane Rubini, Renhai Chen, and Zili Shao. 2018. Emerging NVM: A Survey on Architectural Integration and Research Challenges. *ACM Trans. Design Autom. Electr. Syst.* 23, 2 (2018), 14:1–14:32.
[5] Yanpei Chen et al. 2012. We don't know enough to make a big data benchmark suite-an academia-industry view. *Proc. of WBDB* (2012).
[6] Chaitanya K. Baru et al. 2014. Discussion of BigBench: A Proposed Industry Standard Performance Benchmark for Big Data. In *Proc. of the 6th TPCTC 2014, Hangzhou, China, Sept. 1-5, 2014*.
[7] Lei Wang et al. 2014. BigDataBench: A big data benchmark suite from internet services. In *Proc. of the 20th IEEE HPCA 2014, Orlando, FL, USA, February 15-19, 2014*. IEEE.
[8] Michael Ferdman et al. 2012. Clearing the clouds: a study of emerging scale-out workloads on modern hardware. In *Proc. of the 17th ASPLOS 2012, London, UK, March 3-7, 2012*.
[9] Norman Jouppi et al. 2017. In-Datacenter Performance Analysis of a Tensor Processing Unit. In *Proc. of the 44th ISCA 2017, Toronto, ON, Canada, June 24-28, 2017*.
[10] Peter Mattson et al. 2019. MLPerf Training Benchmark. *CoRR* abs/1910.01500 (2019). arXiv:1910.01500
[11] Vijay Janapa Reddi et al. 2019. MLPerf Inference Benchmark. *CoRR* abs/1911.02549 (2019). arXiv:1911.02549
[12] Zijian Ming et al. 2013. BDGS: A Scalable Big Data Generator Suite in Big Data Benchmarking. In *Proc. of the 2013 Workshop on Big Data Benchmarking, Xi'an, China, July 16-17, 2013 and San José, CA, USA, October 9-10, 2013*.
[13] Flink. 2020. flink.apache.org/.
[14] Ahmad Ghazal, Todor Ivanov, Pekka Kostamaa, Alain Crolotte, Ryan Voong, Mohammed Al-Kateb, Waleed Ghazal, and Roberto V. Zicari. 2017. BigBench V2: The New and Improved BigBench. In *ICDE 2017, San Diego, CA, USA, April 19-22*.
[15] Ahmad Ghazal, Tilmann Rabl, Minqing Hu, Francois Raab, Meikel Poess, Alain Crolotte, and Hans-Arno Jacobsen. 2013. BigBench: Towards An Industry Standard Benchmark for Big Data Analytics. In *SIGMOD 2013* (New York, New York, USA). 1197–1208.
[16] Jim Gray. 1992. *Benchmark Handbook: For Database and Transaction Processing Systems*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
[17] Great Responsibility: The 2018 Big Data Great Power and Matt Turck AI Landscape. 2018. http://mattturck.com/bigdata2018/
[18] Hadoop. 2018. hadoop.apache.org/.
[19] Rui Han, Lizy Kurian John, and Jianfeng Zhan. 2018. Benchmarking Big Data Systems: A Review. *IEEE Trans. Services Computing* 11, 3 (2018), 580–597.
[20] Hive. 2020. hive.apache.org/.
[21] Han Hu, Yonggang Wen, Tat-Seng Chua, and Xuelong Li. 2014. Toward Scalable Systems for Big Data Analytics: A Technology Tutorial. *IEEE Access* 2 (2014), 652–687.
[22] ICT, Chinese Academy of Sciences. 2015. BigDataBench 3.1. http://prof.ict.ac.cn/BigDataBench/
[23] Impala. 2020. impala.apache.org/.
[24] Intel. 2020. HiBench Suite. https://github.com/intel-hadoop/HiBench
[25] Todor Ivanov. 2019. *Classifying, evaluating and advancing big data benchmarks*. Ph.D. Dissertation. Goethe University Frankfurt. http://publikationen.ub.uni-frankfurt.de/frontdoor/index/index/docId/51157
[26] Todor Ivanov, Patrick Bedué, Ahmad Ghazal, and Roberto V. Zicari. 2018. Adding Velocity to BigBench. In *Proc. of the 7th DBTest@SIGMOD 2018, Houston, TX, USA, June 15, 2018*. ACM, 6:1–6:6.
[27] Todor Ivanov, Timo Eichhorn, Arne Berre, Tomás Pariente Lobo, Ivan Martinez Rodriguez, Ricardo Ruiz Saiz, Barbara Pernici, and Chiara Francalanci. 2019. Building the DataBench Workflow and Architecture. (2019).
[28] Todor Ivanov and Rekha Singhal. 2018. ABench: Big Data Architecture Stack Benchmark. In *Companion of the 2018 ACM/SPEC ICPE 2018, Berlin, Germany, April 09-13, 2018*. ACM.
[29] C. Kachris, B. Falsafi, and D. Soudris. 2018. *Hardware Accelerators in Data Centers*. Springer Int.
[30] Alfons Kemper and Thomas Neumann. 2011. HyPer: A hybrid OLTP&OLAP main memory database system based on virtual memory snapshots. In *Proc. of the 27th ICDE 2011, April 11-16, 2011, Hannover, Germany*.
[31] Doug Laney. 2001. 3D data management: Controlling data volume, velocity and variety. *META group research note* 6, 70 (2001), 1.
[32] MADlib. 2020. madlib.apache.org/.
[33] Min Li. 2015. SparkBench. https://bitbucket.org/lm0926/sparkbench
[34] MLlib. 2020. spark.apache.org/mllib/.
[35] Fatma Özcan, Yuanyuan Tian, and Pinar Tözün. 2017. Hybrid Transactional/Analytical Processing: A Survey. In *Proc. of the 2017 ACM SIGMOD 2017, Chicago, IL, USA, May 14-19, 2017*.
[36] Muhammet Mustafa Ozdal. 2018. Emerging Accelerator Platforms for Data Centers. *IEEE Design & Test* 35, 1 (2018), 47–54.
[37] Barbara Pernici, Chiara Francalanci, Angela Geronazzo, Polidori Lucia, Ray Stefano, Riva Leonardo, Arne Jørgen Berre, Ivanov Todor, et al. 2018. Relating Big Data Business and Technical Performance Indicators. In *Conference of the Italian Chapter of AIS*. 1–12.
[38] Nicolas Poggi. 2018. *Microbenchmark*. Springer International Publishing, Cham.
[39] Ivens Portugal, Paulo S. C. Alencar, and Donald D. Cowan. 2018. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Syst. Appl.* 97 (2018).
[40] Tilmann Rabl, Christoph Brücke, Philipp Härtling, Rodrigo Escobar Palacios, Hamesh Patel, Satyam Srivastava, Christoph Boden, Jens Meiners, and Sebastian Schelter. 2019. ADABench-Towards an Industry Standard Benchmark for Advanced Analytics. (2019).
[41] Sherif Sakr, Anna Liu, and Ayman G. Fayoumi. 2013. The family of mapreduce and large-scale data processing systems. *ACM Comput. Surv.* 46, 1 (2013), 11:1–11:44.
[42] Xuanhua Shi, Zhigao Zheng, Yongluan Zhou, Hai Jin, Ligang He, Bo Liu, and Qiang-Sheng Hua. 2018. Graph Processing on GPUs: A Survey. *ACM Comput. Surv.* 50, 6 (2018), 81:1–81:35.
[43] Spark. 2020. spark.apache.org.
[44] SparkSQL. 2020. spark.apache.org/sql/.
[45] SPEC. 2019. https://www.spec.org/.
[46] Tensorflow. 2020. tensorflow.org.
[47] TPC. 2020. www.tpc.org/.
[48] Anuj Vaishnav, Khoa Dang Pham, and Dirk Koch. 2018. A Survey on FPGA Virtualization. In *Proc. of the 28th FPL 2018, Dublin, Ireland, August 27-31, 2018*.
[49] Kizheppatt Vipin and Suhaib A. Fahmy. 2018. FPGA Dynamic and Partial Reconfiguration: A Survey of Architectures, Methods, and Applications. *ACM Comput. Surv.* 51, 4 (2018).
[50] Qingchen Zhang, Laurence T. Yang, Zhikui Chen, and Peng Li. 2018. A survey on deep learning for big data. *Information Fusion* 42 (2018), 146–157.