A survey of scalable deep learning frameworks

Amiri, S.; Salimzadeh, S.; Belloum, A.S.Z.

DOI
10.1109/eScience.2019.00102

Publication date
2019

Document Version
Final published version

Published in
IEEE 15th International Conference on eScience

A Survey of Scalable Deep Learning Frameworks

1st Saba Amiri
Institute of Informatics
University of Amsterdam
Amsterdam, The Netherlands
saba.amiri@student.uva.nl

2nd Sara Salimzadeh
Institute of Informatics
University of Amsterdam
Amsterdam, The Nethelands
sara.salimzadeh@student.uva.nl

3rd A.S.Z Belloum
Institute of Informatics
Univerisity of Amsterdam
Amsterdam, The Netherlands
a.s.z.belloum@uva.nl

***Abstract*—Machine learning models recently have seen a large increase in usage across different disciplines. Their ability to learn complex concepts from the data and perform sophisticated tasks combined with their ability to leverage vast computational infrastructures available today have made them a very attractive choice for many challenges in academia and industry. In this context, deep Learning as a sub-class of machine learning is specifically becoming an important tool in modern computing applications. It has been successfully used for a wide range of different use cases, from medical applications to playing games. Due to the nature of these systems and the fact that a considerable portion of their use-cases deal with large volumes of data, training them is a very time and resource consuming task and requires vast amounts of computing cycles. To overcome this issue, it is only natural to try to scale deep learning applications to be able to run them across in order to achieve fast and manageable training speeds while maintaining a high level of accuracy. In recent years, a number of frameworks have been proposed to scale up ML algorithms to overcome the scalability issue, with roots both in the academia and the industry. With most of them being open source and supported by the increasingly large community of AI specialists and data scientists, their capabilities, performance and compatibility with modern hardware have been honed and extended. Thus, it is not easy for the domain scientist to pick the tool/framework best suited for their needs. This research aims to provide an overview of the relevant, widely used scalable machine learning and deep learning frameworks currently available and to provide the grounds on which researchers can compare and choose the best set of tools for their ML pipeline.**

***Index Terms*—machine learning, deep learning, distributed learning, neural networks, deep networks, deep neural networks, scalability, scalable machine learning**

I. Methodology

The selection criteria for frameworks was aimed at helping e-science researchers review and compare scalable machine learning frameworks. To that end, we defined the selection criteria as scalability, being open source, being in active development, not being redundant(layers on top of other interfaces), and having an API based on one of the common and popular technologies.

II. Frameworks

A. Graphlab

The first graph-based framework expressing iterative algorithms while ensuring sequential consistency and high degree of parallelization. GraphLab [1] maps computational dependencies to data graphs and provides modular scheduling primitives. Computations comprised of local function updates and global synchronization. It is useful for text, video, audio, and image processing.

B. Petuum

Distributed machine learning platform aiming at industrial scale problems using data and model parallel approaches. Petuum [2] is error tolerant and robust against error in intermediate computation stages. Using Dynamic structural policies, it takes changing structural decencies into account. The computations are prioritized via non-uniform convergence approaches. These features make the acceleration from month to days with a lower total cost. However, Petuum lacks automatic recovery.

C. Apache SystemML

A declarative machine learning framework [3] on spark separating algorithm semantics from underlying data representation and run time execution plan. Both scale-up and scale-out options are supported within the framework. Optimizer and runtime components are integrated in SystemML. It also covers wide range of algorithms in different use cases such as classification, clustering, descriptive statistics, matrix factorization, and regression.

D. DeSTIN

One of the first notable proposed scalable frameworks for deep learning based on Bayesian inference and unsupervised learning for dynamic pattern representation [4]. The framework is capable of dealing with high-dimensional input data while modeling the spatiotemporal dependencies in the data via an unsupervised method. The superiority of DeSTIN is that no pre-training is required to initialize the model and the training phase is carried out in a greedylayer-wise manner. It is mostly used for image processing.

E. Nexus

A high-performance parameter orchestration platform lets existing deep learning frameworks scaling out across large clusters on GPUs effectively [5]. With the help of hierarchical and hybrid parameter caching and update frequency adjustment on individual networks, Nexus reduces the communication overhead. In addition to exploiting parameters sparsity,Nexus leverages protocols on high-performance hardware to speed up the training process. Various use cases have been reported for Nexus.

978-1-7281-2451-3/19/$31.00 ©2019 IEEE
DOI 10.1109/eScience.2019.00102

F. ChainerMN

A high-performance distributed deep learning framework which is applicable even in complex use cases like dynamic neural networks, reinforced deep learning [6]. This data parallel model makes use of appropriate libraries to minimize communication time. It has been successfully applied for image processing tasks.

G. Livermore Tournament Fast Batch Learning

A multi-level tournament voting parallel algorithm that deals with the problem of synchronization between workers in a distributed learning scheme by creating a set of models which will be trained separately and periodically will compete with each other in local tournaments [7]. The winning model will continue the training on the local dataset and its metadata and set of hyper-parameters are communicated to the new compute node. The results of running image processing tasks on this framework are available in the literature.

H. Tensorflow

One of the most popular frameworks adopted for ML and DL applications. Several Libraries such as gRPC and CUDA-Aware API are integrated into Tensorflow, enabling training at scale. CUDA-Aware MPI optimization design reduces latency for large massages on GPUs and provide scaling efficiency up to 90% [8]. Many different use cases for this framework have been reported.

I. PyTorch

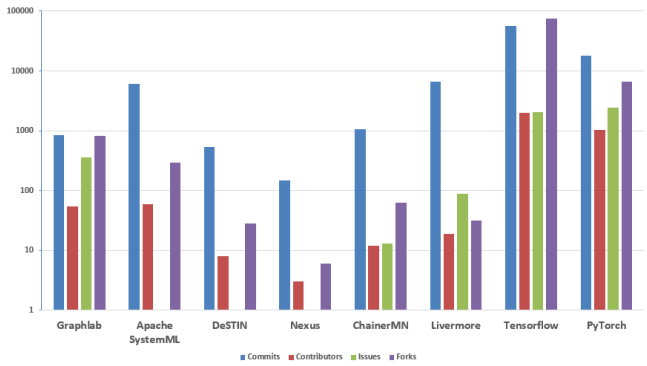
Adeep learning framework developed by Facebook that supports asynchronous distributed training. PyTorch is able to share tensors memory across processes which facilitates Hogwild training and contributes to the scalability of the platform. It has been used for a variety of tasks by the community.

III. Practical Comparison

Figure 1 shows comparison of practical aspects of the covered frameworks on non-linear scale. Information available on github website has been used for this comparison.

Four parameters of *Number of Contributors*, *Number of Commits*, *Number of Issues*, and *Number of Forks* have been selected to evaluate how practical it is to use these frameworks in research work. Number of contributors shows how popular the framework is among active developers in the area of ML/DL. The number of commits shows how active the development of each framework has been-although the number of commits through time would be a better indicator for individual evaluations. The number of issues among other things like the quality of development shows how many active users are employing the framework in their research. The number of forks show how many active developers-outside the core developers- have taken interest in the framework and are using the code base to create and further develop their own systems.

Fig. 1: Practical comparison of ML and DL frameworks



IV. Conclusion

This work was focused on identifying and comparing scalable machine learning and deep learning frameworks relevant for e-science research applications. Having excluded outdated frameworks and frameworks with no active support, we introduce relevant systems and present a brief introduction of each framework, including their key concept and innovation. We also report a high level practical comparison of the frameworks based on their github repository activities, to give users a better idea of the efforts involved in development of each frameworks and their popularity and user support.

References

[1] Y. Low, J. Gonzalez, A. Kyrola, D. Bickson, C. Guestrin, and J. M. Hellerstein, "Graphlab: a new framework for parallel machine learning," in *UAI 2010*, 2010.

[2] E. P. Xing, Q. Ho, W. Dai, J. K. Kim, J. Wei, S. Lee, X. Zheng, P. Xie, A. Kumar, and Y. Yu, "Petuum: A new platform for distributed machine learning on big data," *IEEE Transactions on Big Data*, vol. 1, no. 2, pp. 49–67, June 2015.

[3] M. Boehm, M. W. Dusenberry, D. Eriksson, A. V. Evfimievski, F. M. Manshadi, N. Pansare, B. Reinwald, F. R. Reiss, P. Sen, A. C. Surve, and S. Tatikonda, "Systemml: Declarative machine learning on spark," *Proc. VLDB Endow.*, vol. 9, no. 13, pp. 1425–1436, Sep. 2016. [Online]. Available: https://doi.org/10.14778/3007263.3007279

[4] I. Arel, D. C. Rose, and R. Coop, "Destin: A scalable deep learning architecture with application to high-dimensional robust pattern recognition." in *AAAI Fall Symposium: Biologically Inspired Cognitive Architectures*, 2009.

[5] Y. Wang, L. Zhang, Y. Ren, and W. Zhang, "Nexus: Bringing Efficient and Scalable Training to Deep Learning Frameworks," in *2017 IEEE 25th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Sep. 2017, pp. 12–21.

[6] T. Akiba, K. Fukuda, and S. Suzuki, "Chainermn: Scalable distributed deep learning framework," *CoRR*, vol. abs/1710.11351, 2017.

[7] S. A. Jacobs, N. Dryden, R. Pearce, and B. Van Essen, "Towards Scalable Parallel Training of Deep Neural Networks," in *Proceedings of the Machine Learning on HPC Environments - MLHPC'17*. Denver, CO, USA: ACM Press, 2017, pp. 1–9. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3146347.3146353

[8] A. A. Awan, J. Bédorf, C.-H. Chu, H. Subramoni, and D. K. Panda, "Scalable distributed dnn training using tensorflow and cuda-aware mpi: Characterization, designs, and performance evaluation," *CoRR*, vol. abs/1810.11112, 2018.