

A Machine Learning Solution for Distributed Environments and Edge Computing [†]

Javier Penas-Noce, Óscar Fontenla-Romero  and Bertha Guijarro-Berdiñas * 

Universidade da Coruña, CITIC, Campus de Elviña s/n, 15071 A Coruña, Spain

* Correspondence: berta.guijarro@udc.es

† Presented at the 2nd XoveTIC Conference, A Coruña, Spain, 5–6 September 2019.

Published: 9 August 2019

Abstract: In a society in which information is a cornerstone the exploding of data is crucial. Thinking of the Internet of Things, we need systems able to learn from massive data and, at the same time, being inexpensive and of reduced size. Moreover, they should operate in a distributed manner making use of edge computing capabilities while preserving local data privacy. The aim of this work is to provide a solution offering all these features by implementing the algorithm LANN-DSVD over a cluster of Raspberry Pi devices. In this system, every node first learns locally a one-layer neural network. Later on, they share the weights of these local networks to combine them into a global net that is finally used at every node. Results demonstrate the benefits of the proposed system.

Keywords: machine learning; distributed learning; artificial neural networks; Big Data; privacy-preserving; Internet of Things; edge computing; Raspberry; TensorFlow

1. Introduction

In 2009 Kevin Ashton revised the term “Internet of Things”(IoT), coined by him in 1999, and mentioned that “If we had computers that knew everything there was to know about things—using data they gathered without any help from us—we would be able to track and count everything, and greatly reduce waste, loss and cost” [1]. Nowadays, to be able to take advantage of data in such environments with hyperconnected devices, we need systems with the capacity of operating with massive quantities of data and, at the same time, being inexpensive and of reduced size. Moreover, it is desirable for such systems to operate in a distributed manner making use of their edge computing capabilities, reducing the information flow among devices and preserving at any time the privacy of the data being interchanged as, frequently, they contain sensitive information that can not or should not be shared. The main aim of this work is to provide a solution offering distributed and real time learning capabilities, with privacy-preserving, usable over Raspberry Pi devices of low cost and with limited computing power. Using this system we can built networks of geographically dispersed devices, where each node could learn individually and autonomously using the local data they capture. Later on, they will share their learnt knowledge (that is, the weights of their neural nets) with the other network’s devices so that, between all of them, they can obtain global knowledge similar to that obtained by a centralized system that has been trained using the whole data.

2. Materials and Methods

Traditional training algorithms for neural networks are, in general, iterative, involve high computing times and require human intervention for fine-tuning. Very few of them allow real time (incremental) learning and, even less, allow privacy-preserving distributed learning. As a consequence, they do not adapt well to IoT with edge computing environments. The algorithm LANN-DSVD [2,3] (Linear Artificial Neural Network with Distributed Singular Values Decomposition), developed by the

authors, provides all these desired properties and, for that reason, it will be the one employed in this work. We have implemented this algorithm using the TensorFlow platform and it has been deployed over Raspberry Pi devices. To demonstrate and illustrate the system performance and functioning we will use the MNIST dataset consisting of 28x28 pixels images representing of digits from 0 to 9 divided into a training set (60.000 images) and a test set (10.000 images).

3. Results

In this section, the proposed system is exemplified on a physical environment composed by three Raspberry Pi nodes. A management application allows an administrator, via a web service, to configure the cluster of devices, to activate/deactivate nodes, to force training or knowledge dissemination, among other things. To speed-up deployment and to allow the system scalability we have used Docker as the tool to optimize the launching of new nodes. Optionally, it is also possible to establish a higher power computer as a central node in order to derive the execution of those operations that requires more memory or CPU than the one available at the devices. In this illustrative example, we have used a portable PC as a fourth node but its functioning will be similar to the other devices. It has been included only to demonstrate the possibility of using an heterogeneous cluster. Initially, the MNIST training dataset has been divided in such a way that every node will train using only two or three of the 10 classes. Specifically, node 1 was trained using numbers 0 and 1, node 2 using numbers 3, 5 and 8, node 3 using numbers 2 and 6 and, finally, node 4 was trained with numbers 4, 7 and 9.

Once the nodes have been individually trained, results over the global test set show that, in each node, the sensibility obtained for the classes already included in their training set is always above 90%, while for the rest of classes is 0%, giving a global accuracy between 19.44% and 27.72%. After that, the central node sends a flooding petition to request the nodes for the weight matrices of the neural network every one has obtained using their local data. Later on, these matrices are combined following the LANN-DSVD algorithm and a generalized neural network is obtained that will be spread to every node. Using the same global test set and the generalized model, the new results show that the system is now able to solve the problem recognizing all digits with a global accuracy of 86.06% and a mean sensibility above 85.4% (74% for the worst case—number 5—and 97% for the best case—number 1).

4. Discussion and Conclusions

Distributed learning is an active line of research in order to deal with large and/or distributed data. In this paper, we have presented a physical functional system able to learn from distributed data while maintaining its privacy. It is fast and parameter free, a highly desirable characteristic for large-scale learning environments, where tuning a model can be a very time consuming task, or in situations where autonomous online learning from data streams is required. Although the learnt model is simple (one-layer neural net) it is accurate enough to solve many problems, and in return, it is low computational demanding. This makes the system very suitable for IoT environments. As a future line, we plan to apply the ideas of the LANN-SVD algorithm to deep neural nets.

Funding: This research was funded by the Spanish Secretaría de Estado de Universidades e I+D+i (Grant TIN2015-65069-C2-1-R), Xunta de Galicia (Grants ED431C2018/34, ED341D R2016/045) and FEDER funds.

References

1. Ashton, K. That “Internet of Things” Thing. *RFID J.* **2009**. Available online: <http://www.rfidjournal.com/articles/view?4986> (accessed on 8 August 2019).
2. Fontenla-Romero, O.; Guijarro-Berdiñas, B.; Pérez-Sánchez, B. LANN-SVD: A Non-Iterative SVD-Based Learning Algorithm for One-Layer Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 3900–3905.

3. Fontenla-Romero, O.; Guijarro-Berdiñas, B.; Pérez-Sánchez, B.; Gómez-Casal, M. LANN-DSVD: A privacy-preserving distributed algorithm for machine learning. In Proceedings of the 26th Eur Symp on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges, Belgium, 25–27 April 2018.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).