Thesis Proposal

Student name: Ilias Balampanis **Student ID:** 2014030127

Email: balampanis.ilias@gmail.com

Title

Distributed training of Recurrent Neural networks by FGM protocol.

Keywords

machine learning, recurrent neural networks (RNN), gating units, LSTM cells, geometric monitoring (GM), rebalancing GM, functional geometric monitoring (FGM), rebalancing FGM, distributed training

Abstract

Neural networks are appealing because they learn by example and are strongly supported by statistical and optimization theories. The usage of recurrent neural networks as identifiers and predictors in nonlinear dynamic systems has increased significantly. They can present a wide range of dynamics, due to feedback and are also flexible nonlinear maps. Based on this, there is a need for distributed training on these neural nets, because of the enormous datasets. One of the most known protocols for distributed training is the GM protocol. Our conviction is that this is a very expensive protocol regarding the communication of nodes. Recently, the FGM protocol has tested training on convolutional neural nets. Our goal is to extend this work to RNNs.

Possible issues and solutions

Typically, an RNN is an extremely difficult net to train. Since these nets use backpropagation we once again run into the problem of the vanishing gradient. Unfortunately, the vanishing is exponentially worse for an RNN. The reason for this is that each time step is the equivalent of an entire layer in a feed-forward network. For the same reason, a distributed training model possibly has the same difficulties. If the model is not distributed, the most common solution is to introduce gating units in our system, such as LSTM cells. But, if the training process is distributed, we must find a better safe function than the simple norm as a node communication factor.

Working Plan

1) Construct simple RNN models and train these with multiple datasets, each with different target applications

2) Construct a distributed training model for RNNs using Kamp's GM protocol

3) Measure communication cost on a simulated distributed architecture.

4) Implement a distributed training algorithm for RNNs using FGM/Reb FGM protocol based on the github.com/vsamtuc/ddssim codebase.

5) Evaluate by simulation, as for (3).

Goals

The main goal of this thesis is to prove that FGM/Reb FGM is a better protocol than GM/Reb GM on RNNs training process. So,

- Devise a distributed algorithm on the process of learning of RNNs using GM/Reb GM and FGM/Reb FGM protocol.

- Compare FGM/Reb FGM with Kamp's GM/Reb GM protocols experimentally.

- Hope for a scientific article if this project proves successful.

Timeline of implementation

	Months							
Milestone	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May
Research and bibliography definition	X	X	X	X				
Construct RNN and distributed training models				X	X	X		
Cross-validation, results collection and comparison with related work						X	X	
Thesis Writing							X	X
Presentation								X

References

[1] Ryan R. Curtin et al. “mlpack 3: a fast, flexible machine learning library”. In: *Journal of Open Source Software* 3 (26 2018), p. 726. DOI: `10.21105/joss.00726`. URL: `https://doi.org/10.21105/joss.00726`.

[2] *Detection and Classification of Adult and Fetal ECG Using Recurrent Neural networks , Embedded Volterra and Higher - Order Statistics* . en. OCLC: 1111192923. INTECH Open Access Publisher, 2012. ISBN: 978-953-51-0409-4. URL: `http://www.intechopen.com/articles/show/title/detection-and-classification-of-biomedical-signals-using-embeded-volterra-and-higher-order-statistic` (visited on 12/11/2019).

[3] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. 2nd ed. New York: Wiley, 2001. ISBN: 978-0-471-05669-0.

[4] Barbara Hammer. *Learning with recurrent neural networks*. en. Lecture notes in control and information sciences 253. London ; New York: Springer, 2000. ISBN: 978-1-85233-343-0.

[5] Michael Kamp et al. “Communication- Efficient Distributed Online Prediction by Dynamic Model Synchronization”. en. In: 8724 (2014). Ed. by Toon Calders et al., pp. 623–639. DOI: 10.1007/978-3-662-44848-9_40. URL: http://link.springer.com/10.1007/978-3-662-44848-9_40 (visited on 12/09/2019).

[6] Michael Kamp et al. “Efficient Decentralized Deep Learning by Dynamic Model Averaging”. en. In: *arXiv:1807.03210 [cs, stat]* (Nov. 2018). arXiv: 1807.03210. URL: http://arxiv.org/abs/1807.03210 (visited on 12/09/2019).

[7] Vissarion Konidaris. “Distributed Machine Learning Algorithms via Geometric Monitoring”. en. In: (2019). URL: http://purl.tuc.gr/dl/dias/07D2D80D-F1AC-4937-B5F6-69D82C9A996F.

[8] Vasileios Samoladas. *ddssim*. https://github.com/vsamtuc/ddssim. 2018.

[9] Shai Shalev-Shwartz and Shai Ben-David. “Understanding Machine Learning”. en. In: (), p. 416.