ONLINE LEARNING USING DISTRIBUTED NEURAL NETWORKS

Mauro D. L. Tosi Martin Theobald Vinu E. Venugopal
University of Luxembourg
May 21, 2021

AIM AND SCOPE

This ongoing project explores the usage of **Neural Networks (NN) in an online learning scenario**. NN are machine learning methods that have draught attention in the last decades for their capability to represent and **solve non-convex problems**. However, they are not usually used as part of online learning algorithms, that traditionally can optimise convex functions [1]. This is due to the following common obstacles that online learning encompass.

- input arriving from data-streams,
- concept drift,
- time-sensitivity;

Thus, studying how to overcome those obstacles one may be able to solve non-convex problems in an online learning manner.

OBJECTIVES

Our objective is to design and implement NNs in a distributed setting to reduce the time that it takes to train them, so training can be performed in an Online Learning manner. We plan to accomplish this by completing the following specific objectives.

- define how to **scale out the training** of sparse neural networks,
- propose a framework in which neural networks can adapt themselves based on **concept drift**,
- adapt neural networks training to receive data from **data-streams**,
- develop and distribute **TensAIR**, a framework implementing the above objectives;

By accomplishing this, complex-time-sensitive problems could be appropriately represented, providing more accurate results.

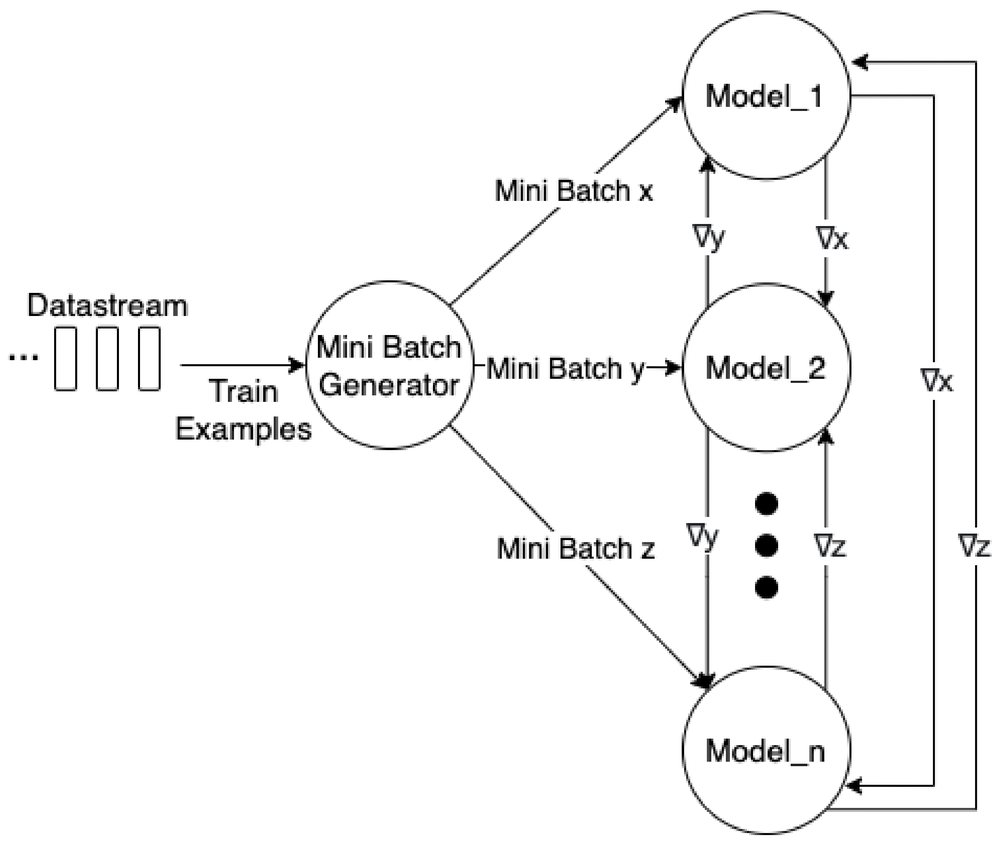
METHODOLOGY

- use online learning batches as NN mini-batches for training on data-streams,
- **distribute** and train replicas of the same NN model **asynchronously** to speedup training,
- exchange gradients calculated among models to ensure a **common convergence**,
- promptly apply received gradients to reduce convergence time,
- resume training when concept drift is detected;

TENSAIR

TensAIR is a distributed online deep learning framework that can train models and use them to make inferences or predictions taking data-streams as input. TensAIR is a TensorFlow [2] framework developed on top of **AIR** [3], a Distributed Dataflow Processing with **Asynchronous Iterative Routing**. This means that it can scale out the training of a deep learning model to multiple nodes with or without GPUs associated with them. To this end, TensAIR uses data parallelism to distribute its computation in a decentralised manner across multiple operators executing asynchronously, as seen in Figure 1.

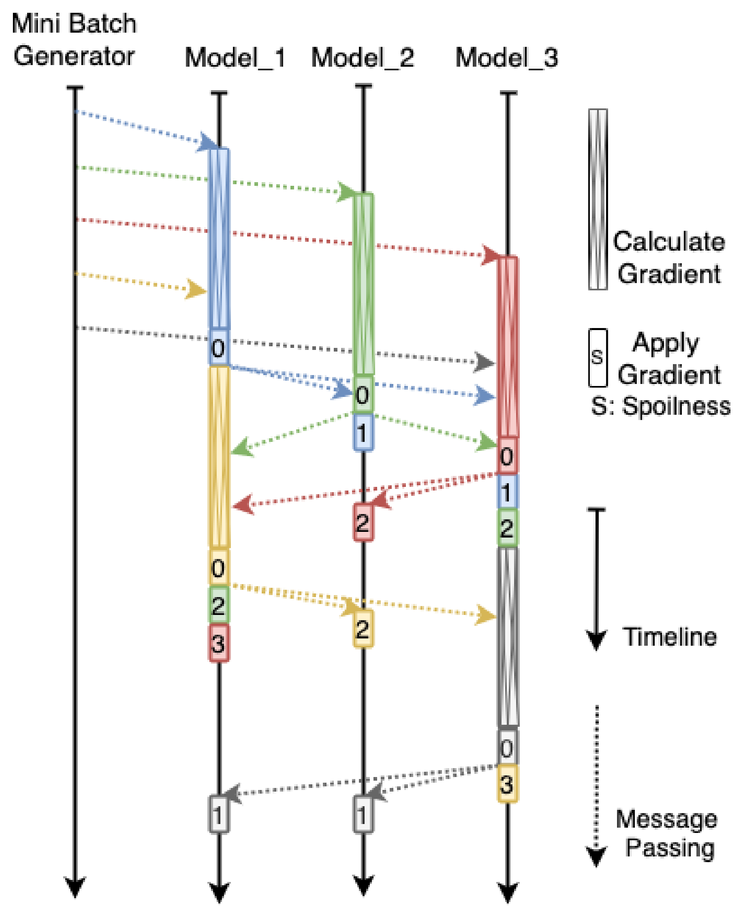
FIGURE 1



TensAIR dataflow with n Models.

Despite TensAIR's asynchronous nature, it is necessary to maintain the *Models* consistent among themselves to guarantee that they will convergence into a common model. This is performed by the exchange of gradients between Model operators, as seen in Figure 2. To ensure that a gradient is not applied to a model too different from the one that it was calculated, it is necessary to control the gradient spoilness. The **gradient spoilness** is the difference between the models that calculated and applied a specific gradient.

FIGURE 2

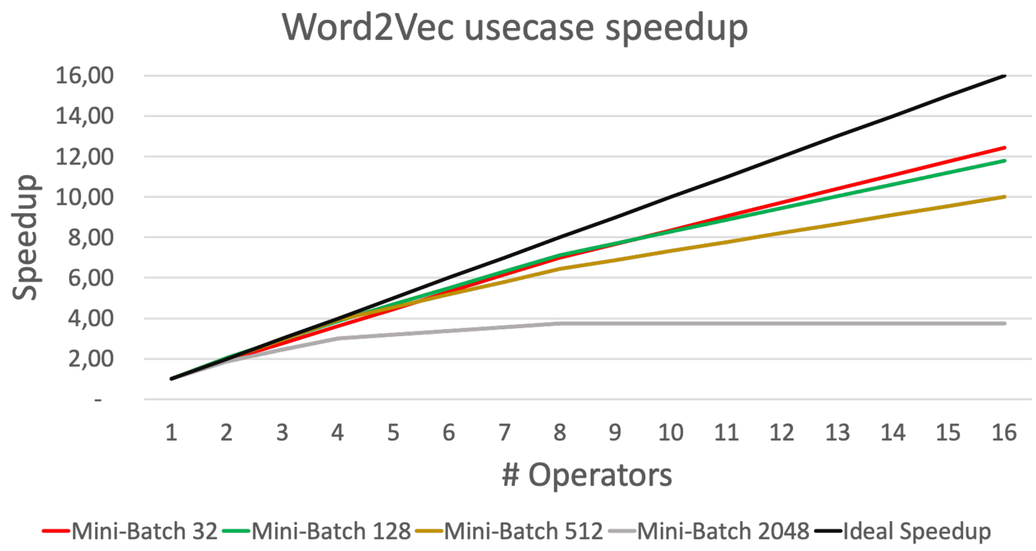


TensAIR Models synchronization.

PRELIMINARY RESULTS

We simulated a "Word Embedding" usecase containing a datastream with training data for a **Skip-gram model**. We used Stochastic Gradient Descent as optimiser, and varied the **mini-batch sizes between 32 and 2,048** over 665,600 training examples. In Figure 3, we observe that **TensAIR achieved a linear speedup** when processing the training examples using smaller mini-batch and varying the number of operators instantiated in a local server.

FIGURE 3



TensAIR speedup in the Word2Vec usecase.

ACKNOWLEDGEMENT

The Doctoral Training Unit **Data-driven computational modelling and applications** (DRIVEN) is funded by the Luxembourg National Research Fund under the PRIDE programme (PRIDE17/12252781). https://driven.uni.lu

REFERENCES

[1] Doyen Sahoo et al. "Online Deep Learning: Learning Deep Neural Networks on the Fly". In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*. International Joint Conferences on Artificial Intelligence Organization, July 2018, pp. 2660–2666. DOI: 10.24963/ijcai.2018/369. URL: https://doi.org/10.24963/ijcai.2018/369.

[2] Martın Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.

[3] Vinu E Venugopal et al. "AIR: A light-weight yet high-performance dataflow engine based on asynchronous iterative routing". In: *2020 IEEE 32nd International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*. IEEE. 2020, pp. 51–58.