A PARALLEL BATCH TRAINING ALGORITHM FOR DEEP NEURAL NETWORK

Yuping Lin

IFLYTEK Laboratory for Neural Computing for Machine Learning

Department of Electrical Engineering and Computer Science

York University, Toronto

October 13, 2015

What does human brain do?

cat

What does human brain do?

$\rightarrow$ $Function$ $\rightarrow$ cat

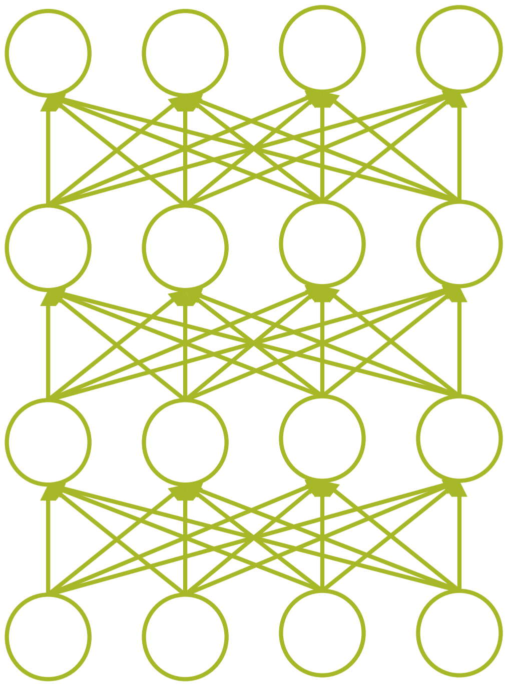
What does neural network do?

Neural Network

cat

Typical structure of NN

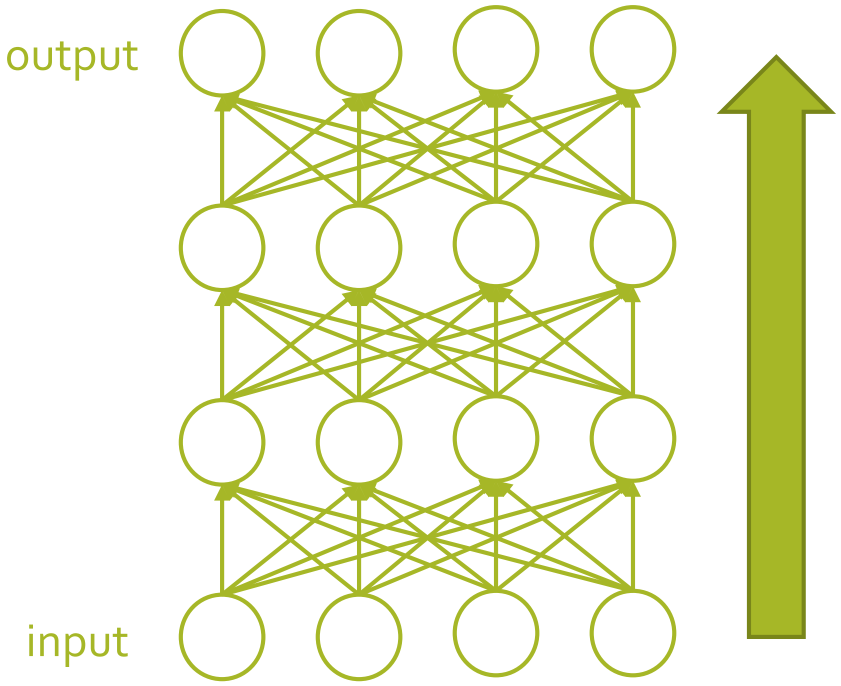
- Has multiple layers;
- Each layer has many units (*a.k.a.* neurons);
- Units are connected by edges;
- Each edge is associated with a weight;



Cycle for NN training

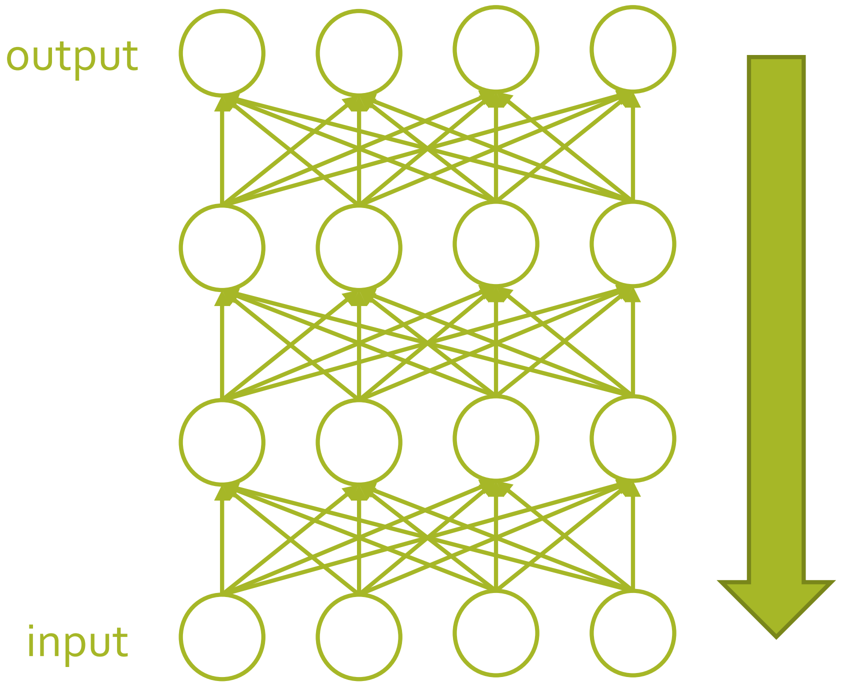
- Forward phase
  - $Z_j^{(l+1)} = F\left(\sum_i w_{ij} \cdot Z_i^{(l)} + b_j\right)$

  Where $F(x)$ is the nonlinear activation function



Cycle for NN training

- Error back propagation
  - $\delta_k^{(out)} = Z_k^{(out)} - T_k^{(out)}$
  - $\delta_i^{(l)} = F'\left(Z_i^{(l)}\right) \cdot \sum_j w_{ij} \cdot \delta_j^{(l+1)}$
  - Where $F'(x)$ is the derivative of the activation function
  - T is the desired output vector
- Weight updating
  - $\Delta w_{ij} = Z_i^{(l)} \cdot \delta_j^{(l+1)}$
  - $w_{ij} = w_{ij} - \gamma \cdot \Delta w_{ij}$
  - Where γ is the learning rate

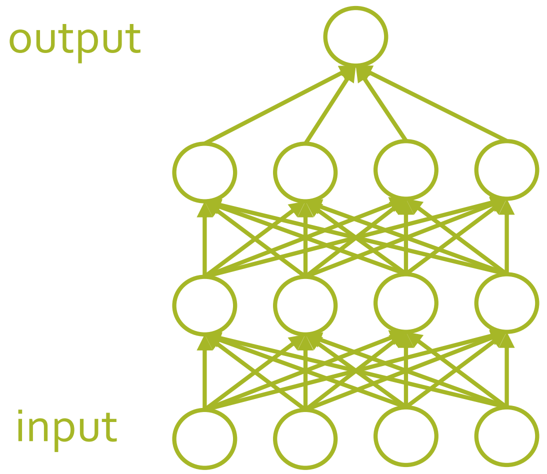


Applications

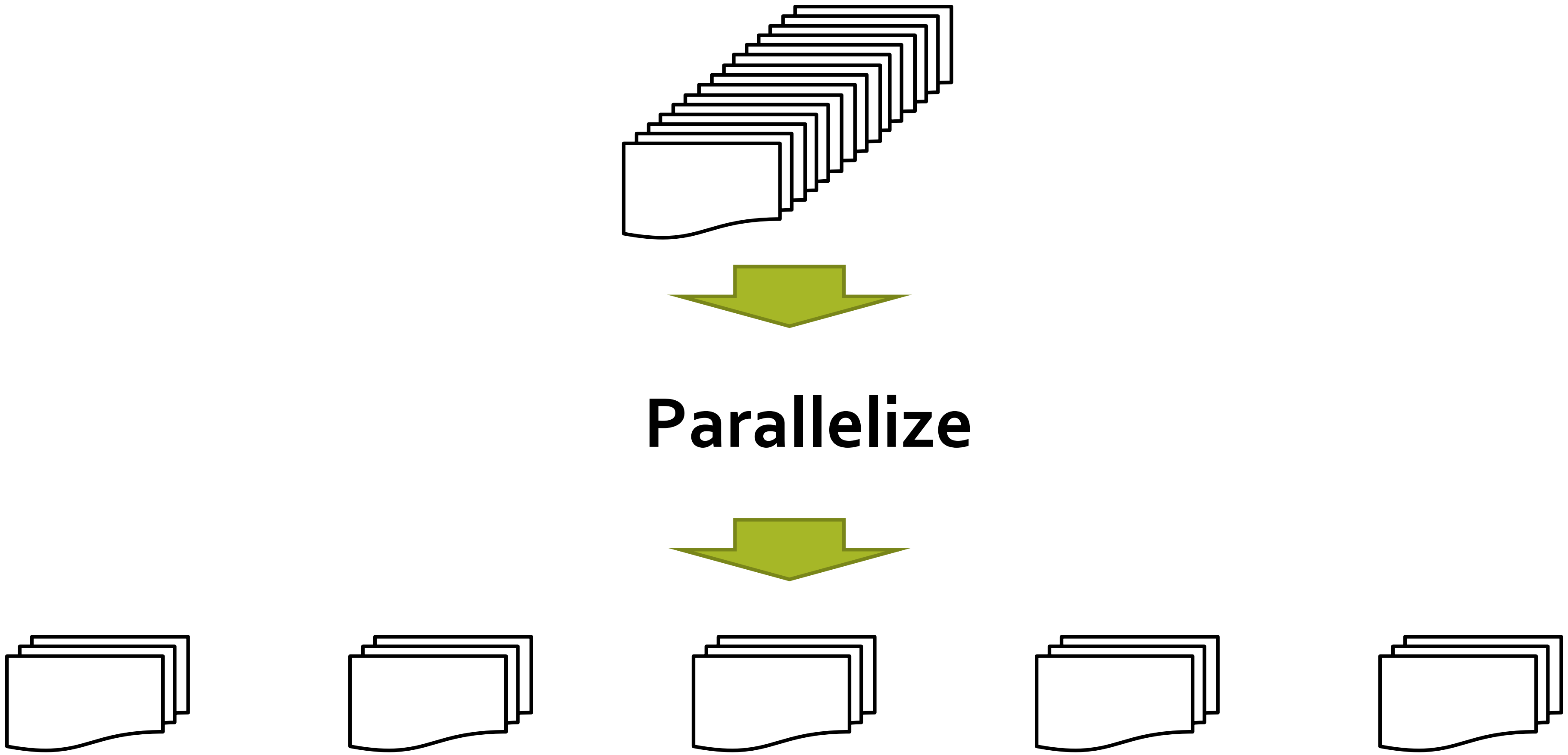
- Computer vision
  - Multi-column DNN, **0.23%** error rate on MNIST (D. Ciresan et al., 2012)
- Speech recognition
  - Bidirectional LSTM, PER **17.7%** on TIMIT (A. Graves et al., 2013)
- Natural Language Processing
  - S-LSTM, **81.9%** accuracy on Stanford Sentiment Treebank (X. Zhu et al. 2015)

Heavy computation load

- Take as example an simple feed-forward NN with 2 hidden layers of size 100. ([100-100-100-1])
  - Has approximately **20,100** parameters;
  - Perform at least **20,100** multiplications in forward phase, for each train sample;
  - Perform at least **40,300** multiplications in error back propagation phase, for each train sample;
  - Plus other operations;



Heavy computation load

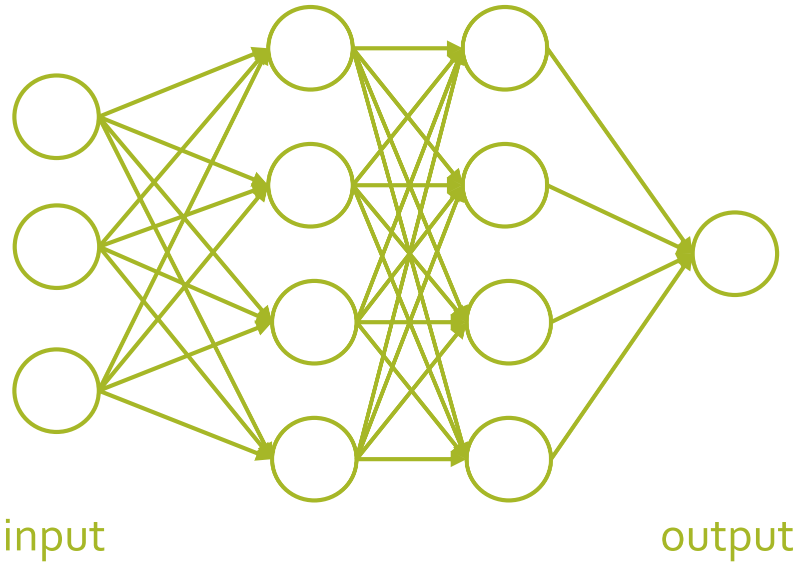


Paper to be presented

- V. Turchenko and V. Golovko. **Parallel batch pattern training algorithm for deep neural network**. *In proceeding of the 2014 International Conference on High Performance Computing Simulation (HPCS)*, pages 697-702, July 2014

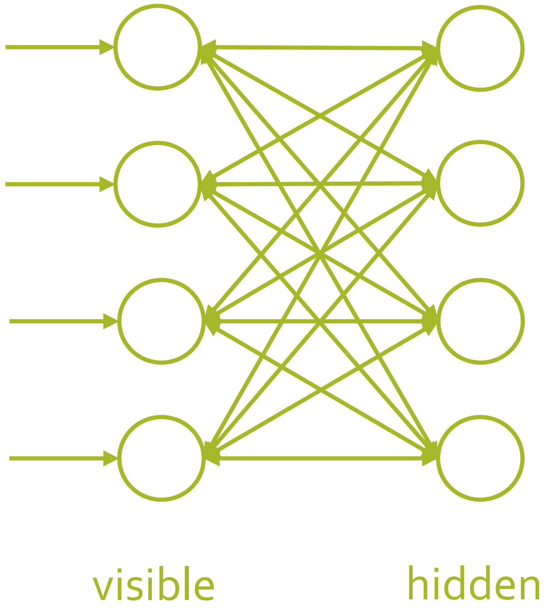
Multi-Layer Perceptron (MLP)

- The authors use an MLP with 2 hidden layers.
- Pre-trained layer by layer with RBMs.
- Then fine-tuned using error back propagation algorithm



Restricted Boltzmann Machine (RBM)

- Has 2 layers of units. Called visible units and hidden units.
- Trained using contrastive divergence algorithm:
  - From inputs $\vec{v}^{(0)}$ compute $\vec{h}^{(1)}$;
  - From $\vec{h}^{(1)}$ compute $\vec{v}^{(1)}$;
  - From $\vec{v}^{(1)}$ compute $\vec{h}^{(2)}$;
  - Compute weight update as: $\Delta w_{ij} = v_i^{(0)} \cdot h_j^{(1)} - v_i^{(1)} \cdot h_j^{(2)}$;
- Use the trained weights to initialize MLP.

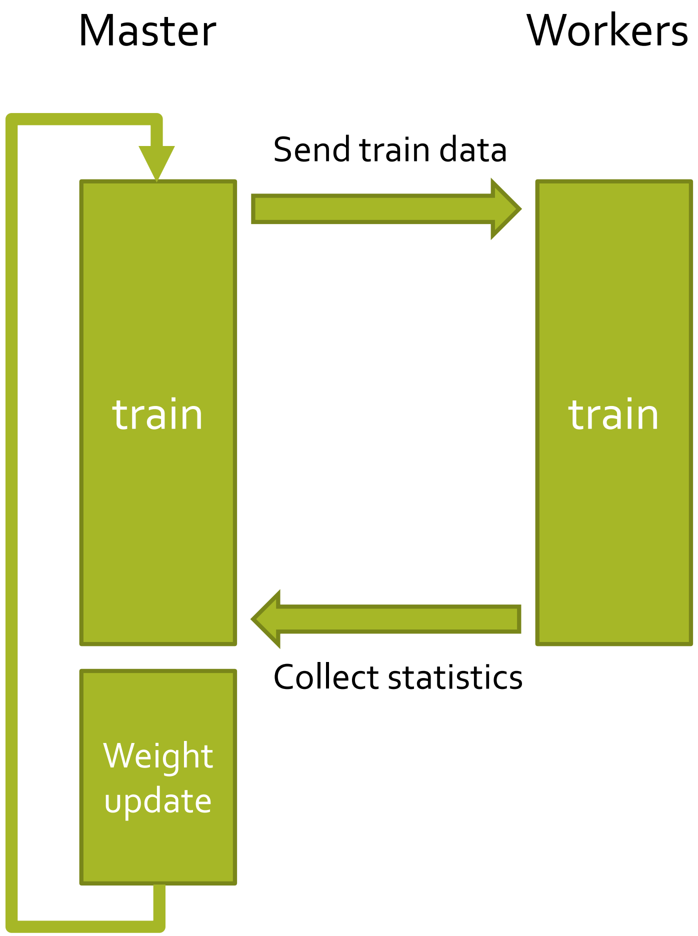


Parallel nature of batch training

- For batch training, weight updates only occur at the end of every batch.
- **For each train sample in the mini-batch:**
  - **Feed input into NN and generate output;**
  - **Back propagate errors;**
  - **Accumulate update statistics.**
- Update weights after the whole mini-batch is traversed.
- The training of each sample within a mini-batch is **independent**.

Parallel batch training

- The proposed parallel batch training algorithm use a single Master thread with many Worker threads.
- Within each mini-batch, the Master first distribute train data to Workers. Then after all the workers finished training, the Master collect training statistics from workers and update weights.



Parallel batch training

- Need synchronization to ensure:
  - Training statistics are collected after and only after all workers finished their training;
  - All Workers start next training iteration only after Master has updated the weights and distributed new training data to them.

Implementation using a monitor

Train: **Monitor**

Begin

Procedure trainSetup (numberOfWorkers : **int**)

Procedure finishTraining ()

Procedure requestUpdate (result : **boolean**)

End

Implementation plan

- Plan:
  - Implement both the sequential and parallel version of the batch training algorithm.
  - Use training time and testing error rate as comparison criteria.

- Challenges:
  - Need parameter tuning;
  - Shorten the waiting time;

References

- A. Graves, A. Mohamed and G. Hinton. **Speech recognition with deep recurrent neural networks**. *In proceeding of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6645-6649, May 2013
- D. Ciresan, U. Meier and J. Schmidhuber. **Multi-column deep neural networks for image classification**. *In proceeding of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3642-3649, June 2012
- X. Zhu, P. Sobhani and H. Guo. **Long short-term memory over recursive structures**. *arXiv preprint arXiv:1503.04881, 2015*