

SIMULATING RNNs on GPUs

SUMMER PROJECT

Zhangsheng Lai

September 27, 2017

MULT-LAYER PERCEPTRON

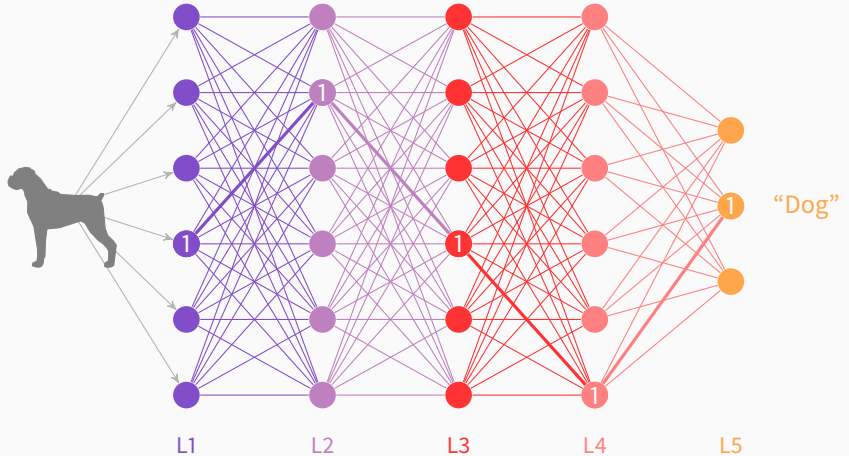
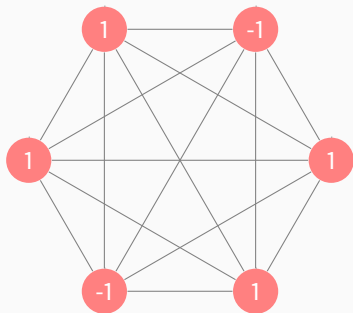


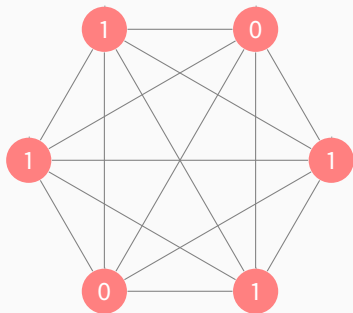
Figure 1: Multi-layer perceptron

HOPFIELD NETWORKS AND BOLTZMANN MACHINES

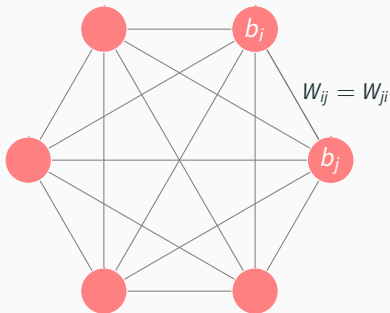
HOPFIELD NETWORKS



HOPFIELD NETWORKS



HOPFIELD NETWORKS

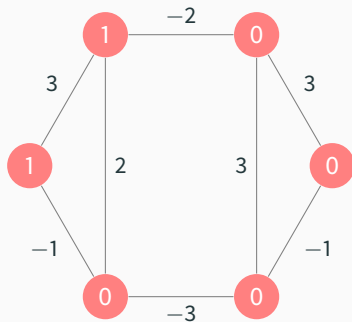


$$\text{Energy configuration, } E = - \sum_{i < j} W_{ij} x_i x_j - \sum_i b_i x_i$$

$$\text{Energy gap, } \Delta E_i = E(x_i = 0) - E(x_i = 1) = \sum_j W_{ij} x_j + b_i$$

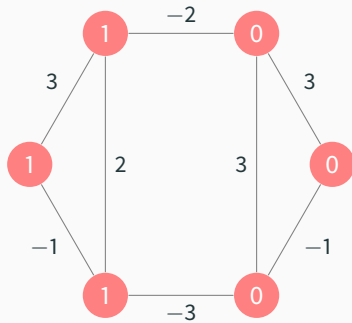
$$\text{Update rule, } x_i := \begin{cases} +1 & \sum_j W_{ij} x_j + b_i \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

HOPFIELD NETWORKS



(1, 0, 0, 0, 0, 1)

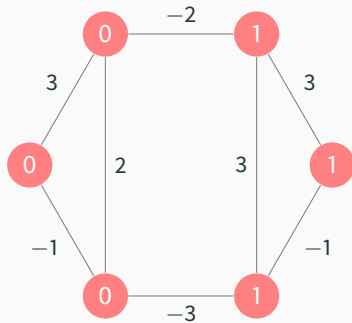
HOPFIELD NETWORKS



(1, 0, 0, 0, 0, 1)

(1, 1, 0, 0, 0, 1)

HOPFIELD NETWORKS

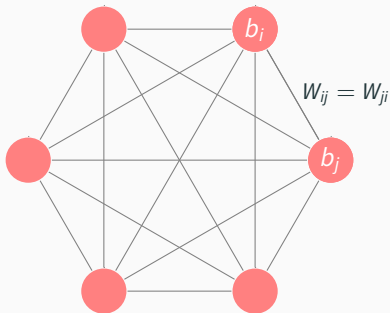


$(1, 0, 0, 0, 0, 1)$

$(1, 1, 0, 0, 0, 1)$

$(0, 0, 1, 1, 1, 0)$

BOLTZMANN MACHINES



$$E = - \sum_{i < j} w_{ij} x_i x_j - \sum_i b_i x_i$$

$$\Delta E_i = E(x_i = 0) - E(x_i = 1) = \sum_j w_{ij} x_j + b_i$$

$$\mathbb{P}(x_i = 1) = \frac{1}{1 + e^{-\Delta E_i / \tau}}$$

BOLTZMANN MACHINES

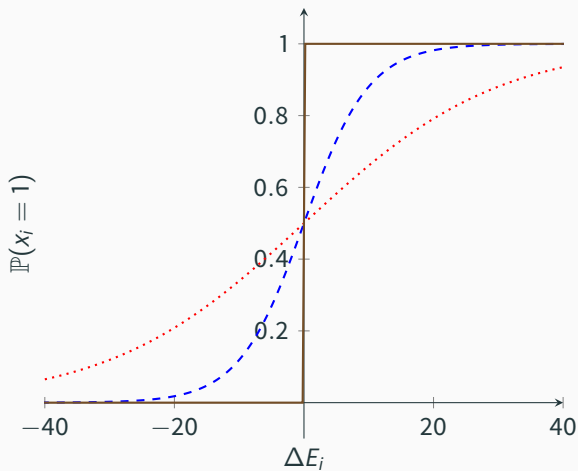
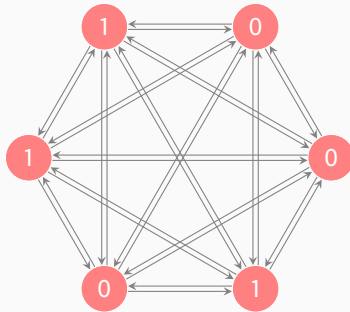


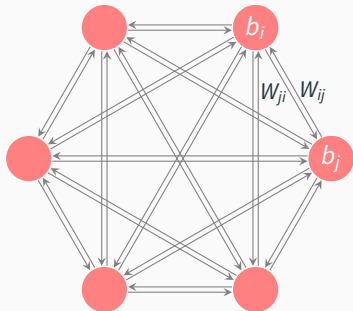
Figure 2: $\tau = 0$ (solid), $\tau = 5$ (dashed), $\tau = 15$ (dotted)

McCULLOCH-PITTS MACHINES

McCULLOCH-PITTS MACHINES



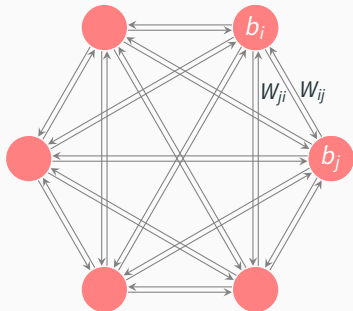
MCCULLOCH-PITTS MACHINES



$$\text{Transition Energy, } E(y, x|\theta) = - \sum_{ji \in E} w_{ji} y_j x_i - \sum_{j \in V} b_j s_j - \sum_{i \in V} b_i s_i$$

$$\Gamma_{yx} = \exp \left(-\frac{1}{2\tau} E(y, x|\theta) + \frac{1}{2\tau} E(x, x|\theta) \right)$$

McCULLOCH-PITTS MACHINES

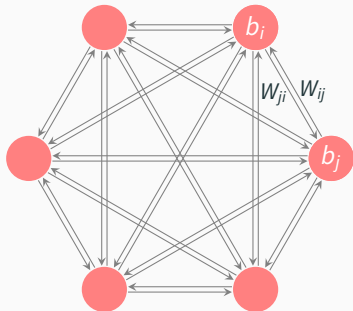


$$\text{Transition Energy, } E(y, x|\theta) = - \sum_{ji \in E} w_{ji} y_j x_i - \sum_{j \in V} b_j s_j - \sum_{i \in V} b_i s_i$$

$$\Gamma_{yx} := \exp \left(\frac{1}{2\tau} s_j z_j \right)$$

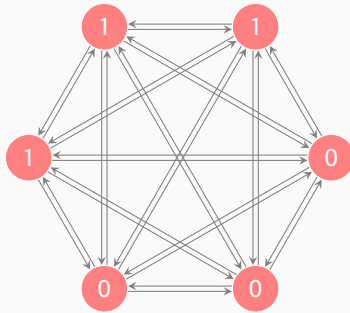
where $s_j = 1 - 2x_j$, $z_j = \sum_i w_{ji} x_i + b_j$ and x, y differ by the j th unit.

McCULLOCH-PITTS MACHINES

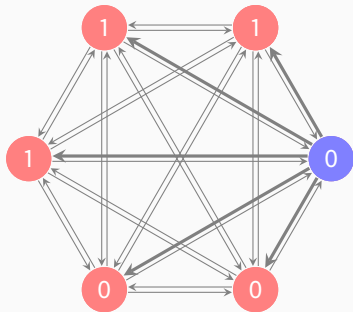


Transition probability from x to y , $p_{yx} = \frac{\lambda_j}{\sum_{j'} \lambda_{j'}}$

McCULLOCH-PITTS MACHINES

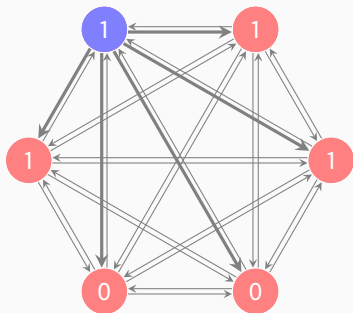


McCULLOCH-PITTS MACHINES



$(T_0, (1, 0, 0, 0, 1, 1))$

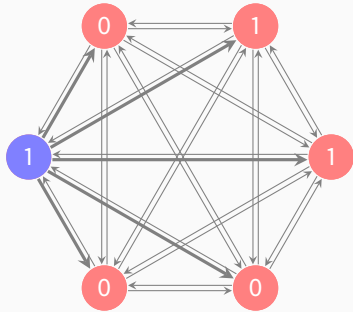
McCULLOCH-PITTS MACHINES



$(T_0, (1, 0, 0, 0, 1, 1))$

$(T_1, (1, 0, 0, 1, 1, 1))$

McCULLOCH-PITTS MACHINES


$$(T_0, (1, 0, 0, 0, 1, 1))$$
$$(T_1, (1, 0, 0, 1, 1, 1))$$
$$(T_2, (1, 0, 0, 1, 1, 0))$$

SIMULATING ON GPUS

SIMULATING ON GPUS

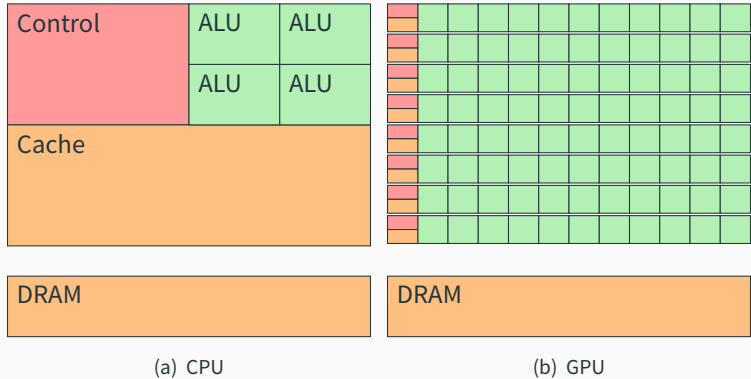
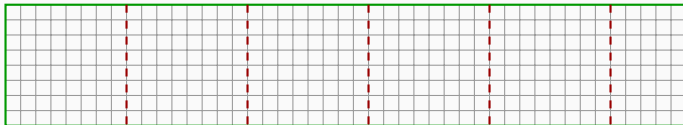


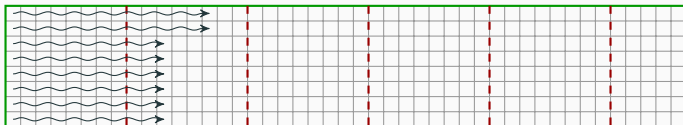
Figure 3: Comparison between the amount of transistors devoted to different functions inside a CPU and a GPU.

SIMULATING ON GPUS

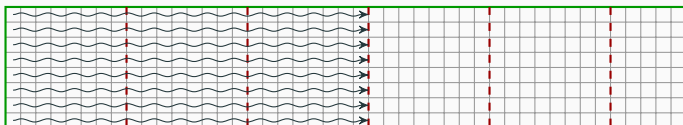
block 0



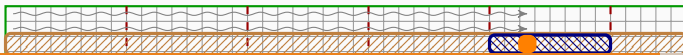
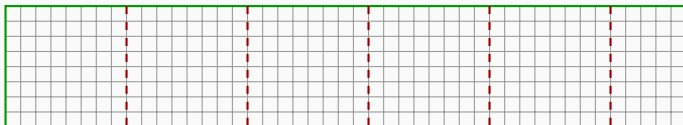
block 1



block 2



block 3



REFERENCES



J. Macor.

A Brief Introduction to Type Theory and the Univalence Axiom

<http://math.uchicago.edu/~may/REU2015/REUPapers/Macor.pdf>



The Univalent Foundations Program

Homotopy Type Theory: Univalent Foundations of Mathematics.

<https://homotopytypetheory.org/book>



The n-Category Café

From Set Theory to Type Theory

https://golem.ph.utexas.edu/category/2013/01/from_set_theory_to_type_theory.html



The nLab

Function Type

<https://ncatlab.org/nlab/show/function+type>