The Power of Extra Analog Neuron

Jiří Šíma

Institute of Computer Science
Academy of Sciences of the Czech Republic

(Artificial) Neural Networks (NNs)

1. mathematical models of biological neural networks

- simulating and understanding the brain (The Human Brain Project)
- modeling cognitive functions

2. computing devices alternative to conventional computers

already first computer designers sought their inspiration in the human brain (e.g., neurocomputer due to `Minsky, 1951`)

- common tools in machine learning or data mining (learning from training data)
- professional software implementations (e.g. `Matlab`, `Statistica` modules)
- successful commercial applications in AI: pattern recognition, control, prediction, decision-making, signal analysis, fault detection, diagnostics, etc.

Neural Networks as Formal Computational Models

- idealized mathematical models of practical NNs (e.g., analog numerical parameters are true real numbers, an unbounded number of computational units, etc.)

- the potential and limits of general-purpose computations with NNs:

  What is ultimately or efficiently computable by particular NN models?

- methodology: the computational power and efficiency of NNs is investigated by comparing formal NN models with more traditional computational models such as finite automata, Turing machines, Boolean circuits, etc.

- NNs may serve as reference models for analyzing alternative computational resources (other than time or memory space) such as analog state, continuous time, energy, temporal coding, etc.

- NN models cover basic characteristics of biological nervous systems (plenty of densely interconnected simple computational units)

  $\longrightarrow$ computational principles of mental processes

The Neural Network Model

- Architecture: s computational units (neurons), indexed as $V = \{1, \ldots, s\}$, connected into a directed graph (V, A) where $A \subseteq V \times V$

- each edge $(i, j) \in A$ from neuron i to j is labeled with a real weight $w_{ji} \in \mathbb{R}$ ($w_{ji} = 0$ iff $(i, j) \notin A$)

- each neuron $j \in V$ is associated with a real bias $w_{j0} \in \mathbb{R}$ (i.e. a weight of $(0, j) \in A$ from an additional neuron $0 \in V$)

- Computational Dynamics: the evolution of network state (output)

$$\mathbf{y}^{(t)} = (y_1^{(t)}, \ldots, y_s^{(t)}) \in [0, 1]^s$$

  at discrete time instant $t = 0, 1, 2, \ldots$

Discrete-Time Computational Dynamics

1. initial state $\mathbf{y}^{(0)} \in [0,1]^s$

2. at discrete time instant $t \geq 0$, an excitation is computed as

$$\xi_j^{(t)} = w_{j0} + \sum_{i=1}^{s} w_{ji} y_i^{(t)} = \sum_{i=0}^{s} w_{ji} y_i^{(t)} \qquad \text{for } j = 1, \ldots, s$$

   where neuron $0 \in V$ has constant output $y_0^{(t)} = 1$ for every $t \geq 0$

3. at the next time instant $t+1$, only the neurons $j \in \alpha_{t+1}$ from a selected subset $\alpha_{t+1} \subseteq V$ update their states:

$$y_j^{(t+1)} = \begin{cases} \sigma_j(\xi_j) & \text{for } j \in \alpha_{t+1} \\ y_j^{(t)} & \text{for } j \in V \setminus \alpha_{t+1} \end{cases}$$

   where $\sigma_j : \mathbb{R} \longrightarrow [0,1]$ is an activation function

Activation Functions

1. binary-state neurons with $y_j \in \{0, 1\}$ (in short, binary neurons)

$$\sigma_H(\xi) = \begin{cases} 1 & \text{for } \xi \geq 0 \\ 0 & \text{for } \xi < 0 \end{cases}$$

Heaviside function

2. analog-state units with $y_j \in [0, 1]$ (briefly analog neurons)

$$\sigma_L(\xi) = \begin{cases} 1 & \text{for } \xi \geq 1 \\ \xi & \text{for } 0 < \xi < 1 \\ 0 & \text{for } \xi \leq 0 \end{cases}$$

saturated-linear function

$$\sigma_S(\xi) = \frac{1}{1 + e^{-\xi}}$$

logistic (sigmoid) function

(Binary) Neural Networks as Language Acceptors

- language (problem) $L \subseteq \{0,1\}^*$ over binary alphabet

- input string $\mathbf{x} = x_1 \ldots x_n \in \{0,1\}^n$ of arbitrary length $n \geq 0$ is sequentially presented, bit after bit, via an input neuron $1 \in V$,

$$y_1^{(d(\tau-1))} = x_\tau \qquad \text{for microscopic time } \tau = 1, \ldots, n$$

  where integer $d \geq 1$ is the time overhead for processing a single input bit

- output neuron $2 \in V$ signals whether $x \overset{?}{\in} L$,

$$y_2^{(dn)} = \begin{cases} 1 & \text{for } x \in L \\ 0 & \text{for } x \notin L \end{cases}$$

  (at time $T(n)$ for analog networks)

Computational Power of Neural Networks

(with the saturated-linear activation function)

depends on the information contents of weight parameters:

1. integer weights: finite automaton (Minsky, 1967)

2. rational weights: Turing machine (Siegelmann, Sontag, 1995)
   polynomial time $\equiv$ complexity class P

3. arbitrary real weights: "super-Turing" computation (Siegelmann, Sontag, 1994)
   polynomial time $\equiv$ nonuniform complexity class P/poly
   exponential time $\equiv$ any I/O mapping

Computational Power of Neural Networks

(with the saturated-linear activation function)

depends on the information contents of weight parameters:

1. integer weights: finite automaton (Minsky, 1967)

2. rational weights: Turing machine (Siegelmann, Sontag, 1995)

   polynomial time $\equiv$ complexity class P

   polynomial time & increasing Kolmogorov complexity of real weights $\equiv$ a proper hierarchy of nonuniform complexity classes between P and P/poly

   (Balcázar, Gavaldà, Siegelmann, 1997)

3. arbitrary real weights: "super-Turing" computation (Siegelmann, Sontag, 1994)

   polynomial time $\equiv$ nonuniform complexity class P/poly

   exponential time $\equiv$ any I/O mapping

Computational Power of Neural Networks

(with the saturated-linear activation function)

depends on the information contents of weight parameters:

1. integer weights: finite automaton (Minsky, 1967)

   a gap with respect to the Chomsky hierarchy ???

   regular (Typ-3) $\times$ recursively enumerable (Type-0) languages

2. rational weights: Turing machine (Siegelmann, Sontag, 1995)

   polynomial time $\equiv$ complexity class P

   polynomial time & increasing Kolmogorov complexity of real weights $\equiv$ a proper hierarchy of nonuniform complexity classes between P and P/poly

   (Balcázar, Gavaldà, Siegelmann, 1997)

3. arbitrary real weights: "super-Turing" computation (Siegelmann, Sontag, 1994)

   polynomial time $\equiv$ nonuniform complexity class P/poly

   exponential time $\equiv$ any I/O mapping

A Binary Neural Network with an Analog Neuron (NN1A)

two analog neurons (together with a few binary ones) can implement a 2-stack pushdown automaton $\equiv$ Turing machine `(Siegelmann, Sontag, 1995)`

$\longrightarrow$ **What is the computational power of ONE extra analog neuron ?**

- rational weights: $w_{ji} \in \mathbb{Q}$ for every $j, i \in V$
- binary-state neurons $j \in V \setminus \{s\}$ employ the Heaviside activation function:
  $$\sigma_j = \sigma_H \quad \longrightarrow \quad y_j \in \{0,1\} \quad \text{for every} \quad j = 0, \ldots, s-1$$
- an extra analog-state unit $s \in V$ applies the saturated-linear activation function:
  $$\sigma_s = \sigma_L \quad \longrightarrow \quad y_s \in \mathbb{I} = \mathbb{Q} \cap [0,1]$$

Theorem 1 *A binary-state neural network with an analog neuron is (by mutual simulations) computationally equivalent to a finite automaton with a register.*

A Finite Automaton with a Register (FAR)

(reminiscent of a FA with multiplication due to `Ibarra, Sahni, Kim, 1976`)

a nine-tuple $(Q, \Sigma, \{I_1, \ldots, I_p\}, a, (\Delta_1, \ldots, \Delta_p), \delta, q_0, z_0, F)$ where

- Q is a finite set of states including start state $q_0 \in Q$ and subset of accept (final) states $F \subseteq Q$; $\Sigma = \{0, 1\}$ is a binary input alphabet
- automaton is augmented with a register storing a rational number from domain $\mathbb{I} = \mathbb{Q} \cap [0, 1]$ which is partitioned into intervals $\mathbb{I} = I_1 \cup I_2 \cup \ldots \cup I_p$ of different types: open, closed, half-closed, or degenerate (containing a single point)
- each interval I_r $(1 \leq r \leq p)$ is associated with a local state-transition function $\delta_r : Q \times \Sigma \longrightarrow Q$ which is employed if the current register value falls into I_r
- the register is initialized with start value $z_0 \in \mathbb{I}$ and its value $z \in I_r$ is updated to $\sigma_L\left(az + \Delta_r(q, x)\right)$ where $a \in \mathbb{Q}$ is a multiplier and $\Delta_r : Q \times \Sigma \longrightarrow \mathbb{Q}$ $(1 \leq r \leq p)$ is a rational shift function depending on current state and input bit
- $\delta : Q \times \mathbb{I} \times \Sigma \longrightarrow Q \times \mathbb{I}$ is a global state-transition function which, given current state $q \in Q$, register value $z \in \mathbb{I}$ and input bit $x \in \Sigma$, produces the new state and register value as

$$\delta(q, z, x) = (\delta_r(q, x), \sigma_L(az + \Delta_r(q, x))) \quad \text{if} \quad z \in I_r$$

What Is the Computational Power of NN1A $\equiv$ FAR ?

upper bound: a finite automaton with a register can be simulated by a (deterministic) linear bounded automaton (linear-space Turing machine) $\longrightarrow$

neural networks with one analog unit accept at most context-sensitive languages

? lower bound: Is there a language accepted by a FAR which is not context-free ?

the computational power of NN1A is between Type-3 and Type-1 languages in the Chomsky hierarchy

When the extra analog neuron does not increase the power of binary NNs ?

Quasi-Periodic Power Series

a power series $\sum_{k=0}^{\infty} b_k a^k$ is eventually quasi-periodic if there is a real number P and an increasing infinite sequence of indices $0 \leq k_1 < k_2 < k_3 < \cdots$ such that $m_i = k_{i+1} - k_i$ is bounded and for every $i \geq 1$,

$$\frac{\sum_{k=0}^{m_i-1} b_{k_i+k} a^k}{1 - a^{m_i}} = P$$

- example: eventually periodic sequence $(b_k)_{k=0}^{\infty}$ where k_1 is the length of preperiodic part and $m_i = m$ is the period

- for $|a| < 1$, the sum of eventually quasi-periodic power series is

$$\sum_{k=0}^{\infty} b_k a^k = \sum_{k=0}^{k_1-1} b_k a^k + a^{k_1} P \qquad \text{(equals } P \text{ for } k_1 = 0)$$

- the sum does not change if any quasi-repeating block $b_{k_i}, b_{k_i+1}, \ldots, b_{k_{i+1}-1}$ is removed or inserted in between two other blocks, or if the blocks are permuted

- Let $k_1 > 1$, $|a| < 1$, and $a = \frac{a_1}{a_2}$, $c = \frac{c_1}{c_2}$ be irreducible. If $c = \sum_{k=0}^{\infty} b_k a^k \in \mathbb{I}$ is eventually quasi-periodic, then $a_1 \,|\, (c_2 b_0 - c_1)$ and for every $i, j \geq 1$, $a_2 \,|\, c_2(b_{k_i-1} - b_{k_j-1})$ (e.g. $a_2 \,|\, c_1$ for $k_1 = 0$).

The Main Result

Theorem 2 *Let* $R = (Q, \Sigma, \{I_1, \ldots, I_p\}, a, (\Delta_1, \ldots, \Delta_p), \delta, q_0, z_0, F)$ *be a finite automaton with a register satisfying* $|a| \leq 1$*. Denote by* $C \subseteq \mathbb{I}$ *the set of all endpoints of intervals* $I_1, \ldots, I_p$ *and* $B = \bigcup_{r=1}^{p} \Delta_r(Q \times \Sigma) \cup \{0, 1, z_0\} \subseteq \mathbb{Q}$ *includes all possible shifts. If every series* $\sum_{k=0}^{\infty} b_k a^k \in C$ *with all* $b_k \in B$ *is eventually quasi-periodic, then* $L(R)$ *accepted by* R *is a regular language.*

Open Problems:

- complete the analysis for $|a| > 1$
- a necessary condition for $L(R)$ to be regular

Expansions of Numbers in Non-Integer Bases (Rényi, 1957)

- a power series $c=\sum_{k=1}^{\infty} b_k a^k$ can be interpreted as a so-called β-expansion of $c\in[0,1]$ in base $\beta=\frac{1}{a}>1$ using the digits b_k from a finite set B

- any $c\in\left[\frac{\min B}{\beta-1},\frac{\max B}{\beta-1}\right]$ has a β-expansion iff

$$\max_{b,b'\in B\,;\,b\neq b'} |b-b'| \leq \frac{\max B - \min B}{\beta - 1}$$

  i.e. any $c\in\left[0,\frac{\lceil\beta\rceil-1}{\beta-1}\right]$ has a β-expansion for usual $B=\{0,1,\ldots,\lceil\beta\rceil-1\}$

  $\times$ for $B=\{0,1\}$ and $\beta>2$, i.e. $0<a<\frac{1}{2}$, there are $c\in[0,1]$ with no β-expansions (β-expansions create a Cantor-like set)

  $\longrightarrow$ a FAR accepts a regular language if the endpoints of intervals $I_1,\ldots,I_p$ do not have β-expansions at all

- for integer bases $\beta\in\mathbb{Z}$ and $B=\{0,\ldots,\beta-1\}$, a β-expansion of c is eventually periodic (or finite) iff c is a rational number

  $\longrightarrow$ a FAR with multiplier $a=1/\beta$, where $\beta\in\mathbb{Z}$, accepts a regular language

Unique β-Expansions (Sidorov, 2009)

for simplicity, assume $B = \{0, 1\}$ and $1 < \beta < 2$, that is, $\frac{1}{2} < a < 1$

- $\beta \in (1, \varphi)$ where $\varphi = (1 + \sqrt{5})/2$ is the *golden ratio*, $0.618\ldots \le a < 1$: any $c \in (0, 1]$ has a continuum of distinct β-expansions including non-quasi-periodic ones

- $\beta \in [\varphi, q_c)$ where q_c is the *Komornik-Loreti constant*, $0.559\ldots < a \le 0.618\ldots$: countably many $c \in [0, 1]$ have unique eventually periodic β-expansions

  (q_c is the unique solution of equation $\sum_{k=1}^{\infty} t_k q_c^{-k} = 1$ where $(t_k)_{k=1}^{\infty}$ is the *Thue-Morse sequence* in which $t_k \in \{0, 1\}$ is the parity of the number of 1's in the binary representation of k)

- $\beta \in [q_c, 2)$, i.e. $\frac{1}{2} < a \le 0.559\ldots$: a continuum of numbers $c \in [0, 1]$ have unique β-expansions which create a Cantor-like set

Quasi-Periodic Greedy β-Expansions (Schmidt, 1980; Hare, 2007)

the lexicographically maximal (resp. minimal) β-expansion of a number is called greedy (resp. lazy)

Per$(\beta) \subseteq [0,1]$ is a set of numbers having quasi-periodic greedy β-expansions

Theorem: If $\mathbb{I} = \mathbb{Q} \cap [0,1] \subseteq$ Per(β), then β is either a Pisot or a Salem number.

Pisot (resp. *Salem*) number is a real algebraic integer (a root of some monic polynomial with integer coefficients) > 1 such that all its Galois conjugates (other roots of such a polynomial with minimal degree) are in absolute value < 1 (resp. ≤ 1 and at least one $= 1$)

Theorem: If β is a Pisot number, then $\mathbb{I} \subseteq$ Per(β). (for Salem numbers still open)

Corollary: For any non-integer rational $\beta \in \mathbb{Q} \setminus \mathbb{Z}$ there exists $c \in \mathbb{I}$ whose (greedy) β-expansion is not quasi-periodic.

Conclusions

- the analysis of computational power of neural nets between integer a rational weights has partially been refined

- our preliminary study reveals open problems and new research directions

- an interesting link to an active research field on β-expansions of numbers in non-integer bases:

  Adamczewski et al., 2010; Allouche et al., 2008; Chunarom & Laohakosol, 2010; Dajani et al., 2012; De Vries & Komornik, 2009; Glendinning & Sidorov, 2001; Hare, 2007; Komornik & Loreti, 2002; Parry, 1960; Rényi, 1957; Schmidt, 1980; Sidorov, 2009; & references there