Symbolic vs. subsymbolic representation in cognitive science and artificial intelligence

Vladimír Kvasnička
FIIT STU

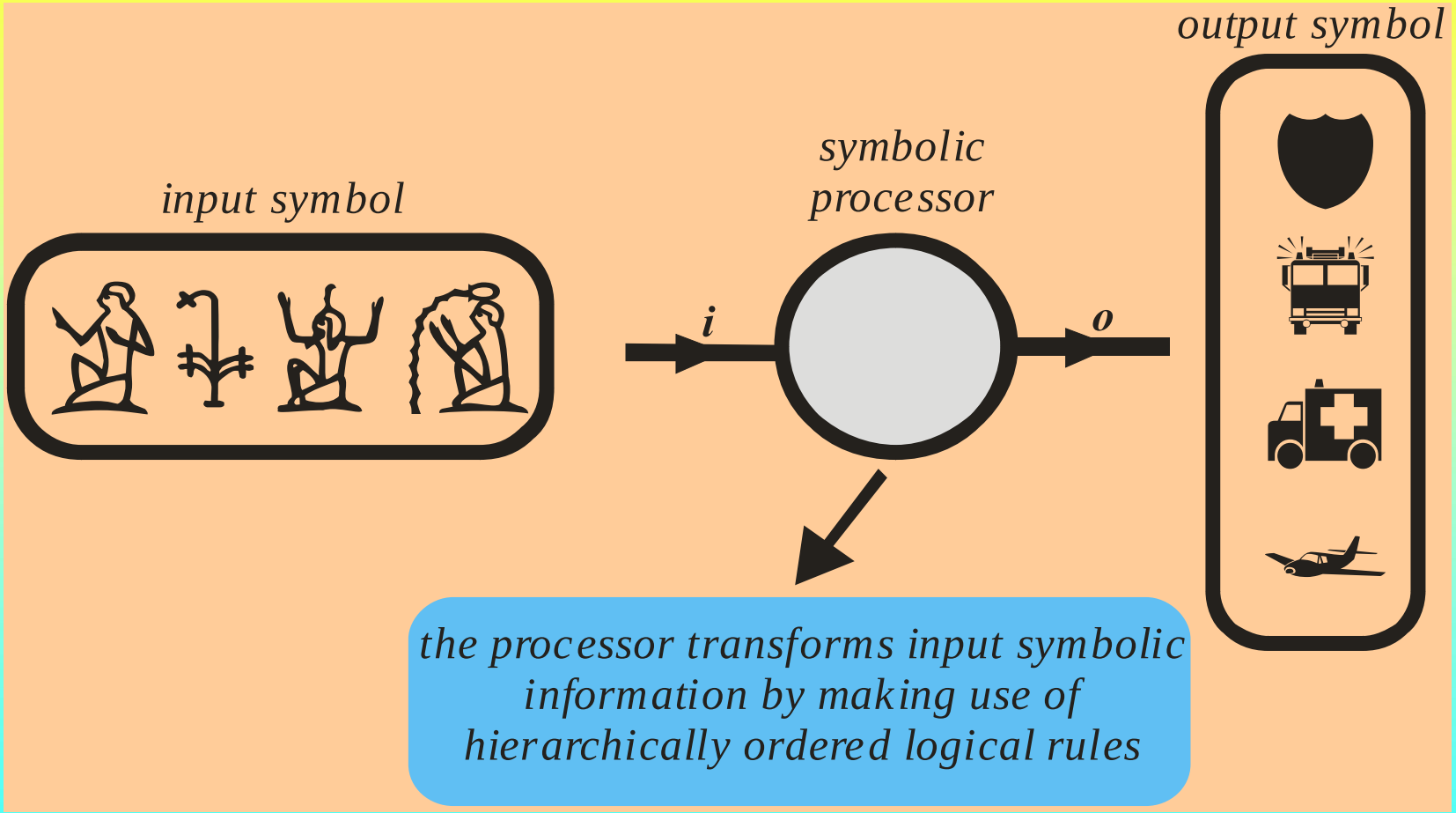
1. Classical (symbolic) artificial intelligence

Basic problem of classical artificial intelligence (AI):

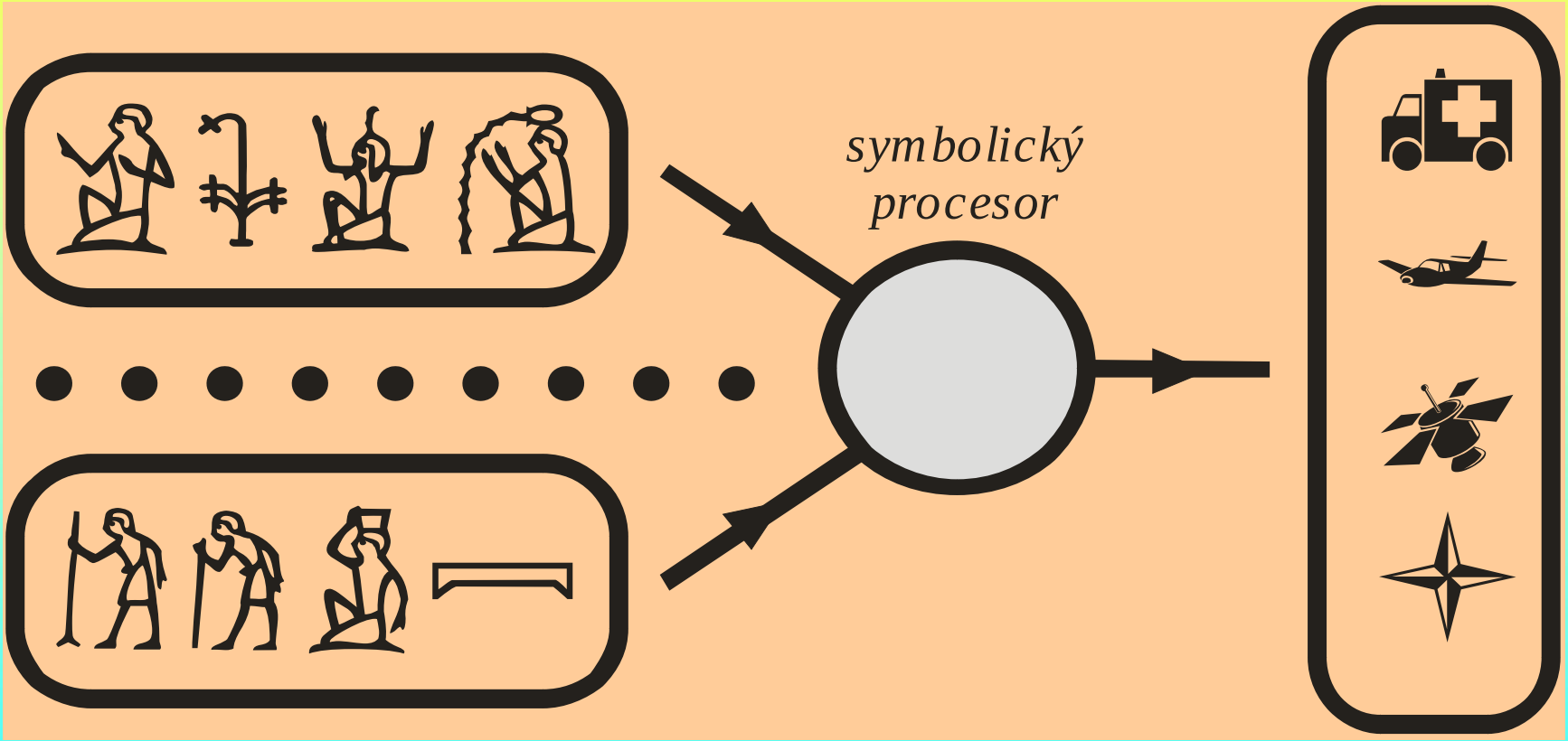
(1) knowledge representation,
(2) reasoning processes,
(3) problem solving,
(4) communication in natural language,
(5) robotics,
(6) ….

are solved in the framework by the so-called ***symbolic representation***. Its main essence consists that for given elementary problems we have available symbolic processors, which on their input site accept symbolic input information and on their opposite output site create symbolic output information.

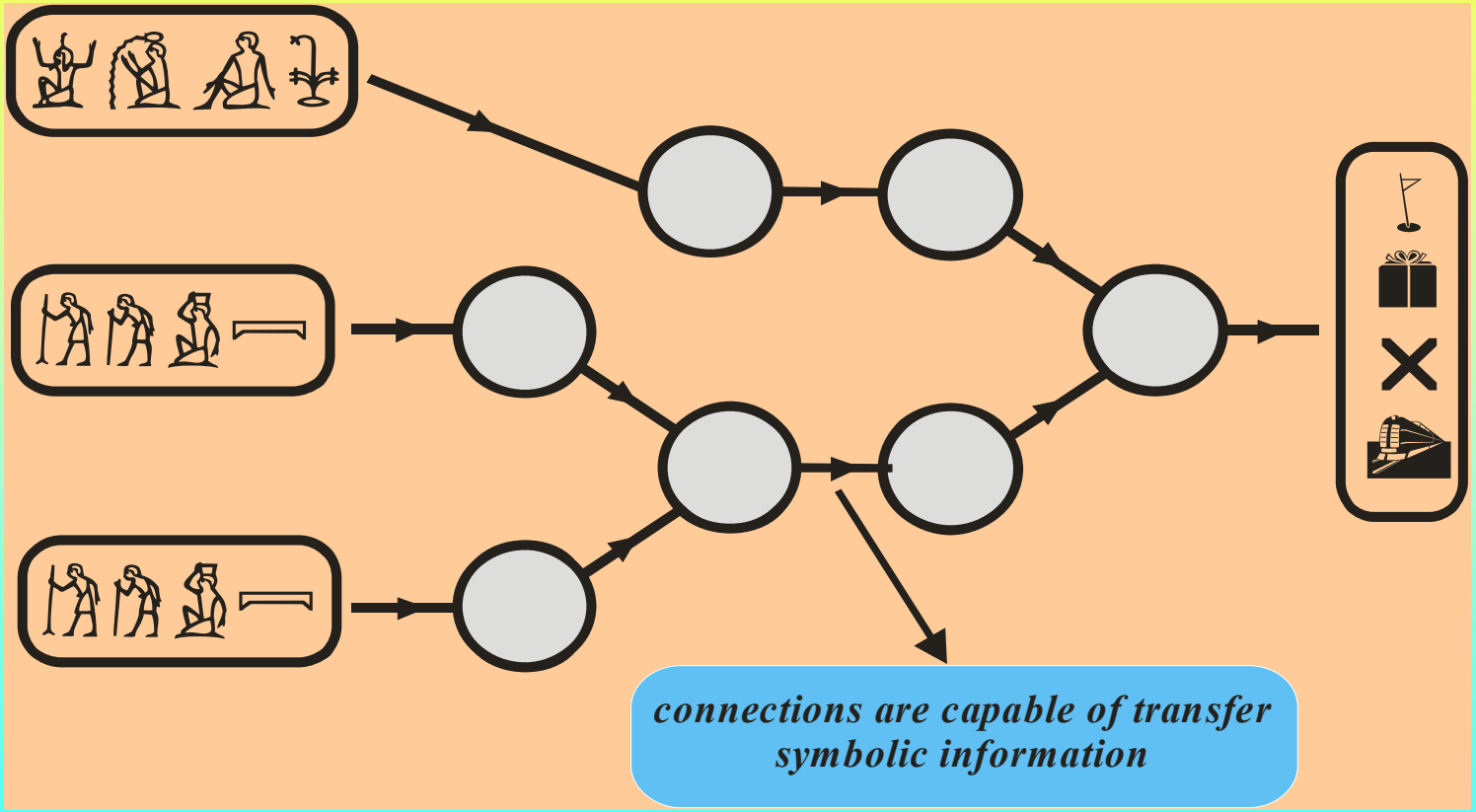
Elementary symbolic processor



More complex symbolic processor (with two more input channels)



Network of symbolic processors
(*symbolic network*)

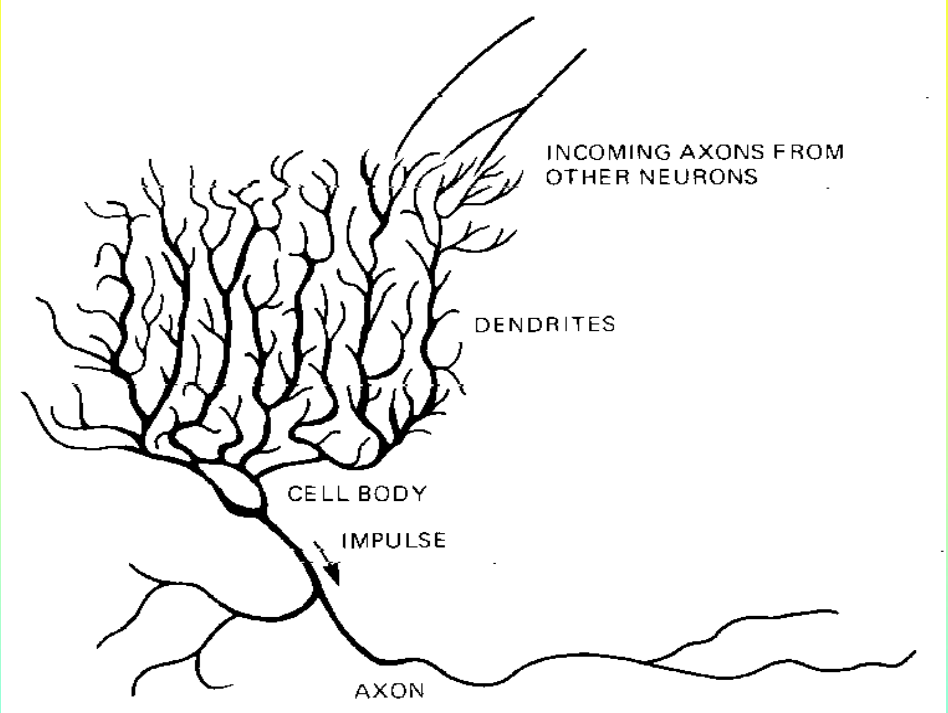


2. Subsymbolic artificial intelligence

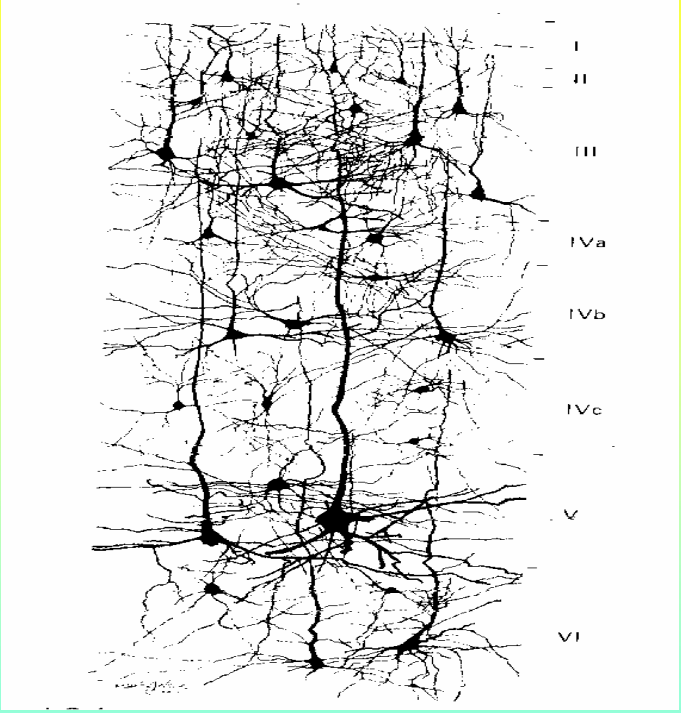
(*machine intelligence – neural networks*)

In subsymbolic (connectionist) theory information is parallelly processed by simple calculations realized by neurons. In this approach information is represented by a simple sequence pulses. Subsymbolic models are based on a metaphor human brain, where cognitive activities of brain are interpreted by theoretical concepts that have their origin in neuroscience:

(1) neuron received information from its neighborhood of other neurons,
(2) neuron processes (integrated) received information,
(3) neuron sends processed information other neurons from its neighborhood.



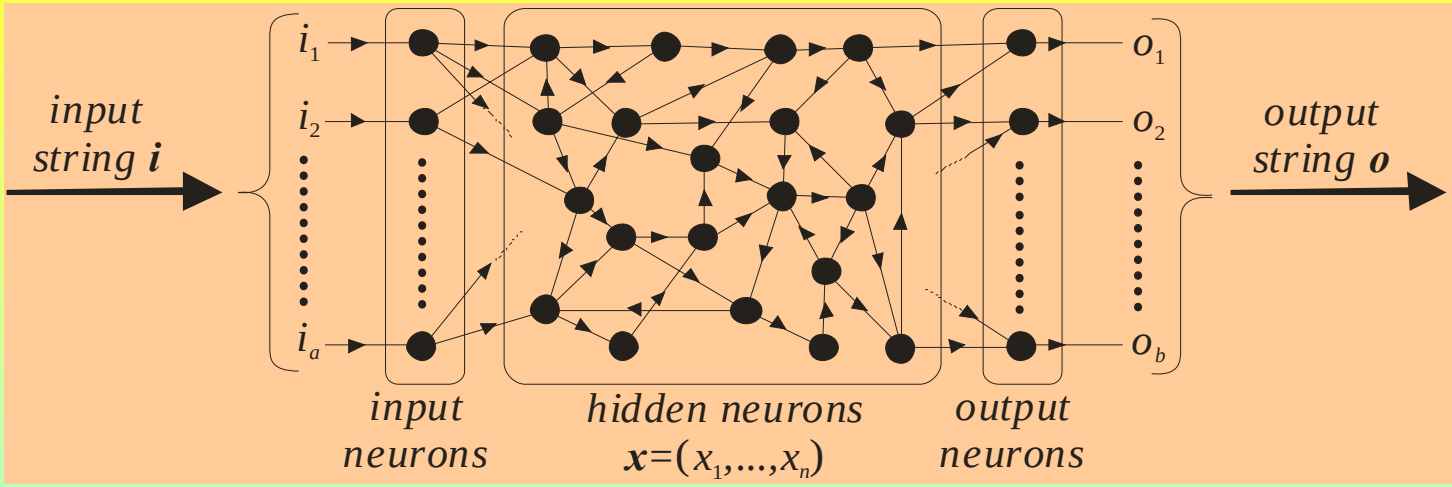
A



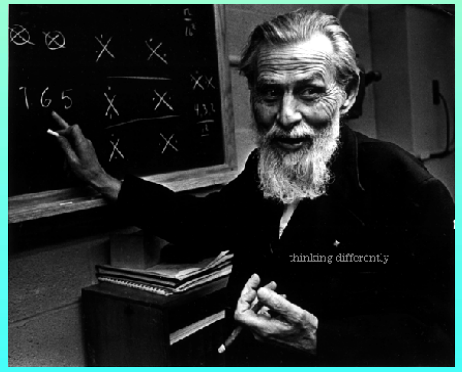
B

Diagram A corresponds to the neuron (nerve cell) composed of many incoming connections (dendrites) and outgoing not very branched connection (axon). Diagram B shows neurons in the brain that are highly interconnected.

Subsymbolic network – *neural network*

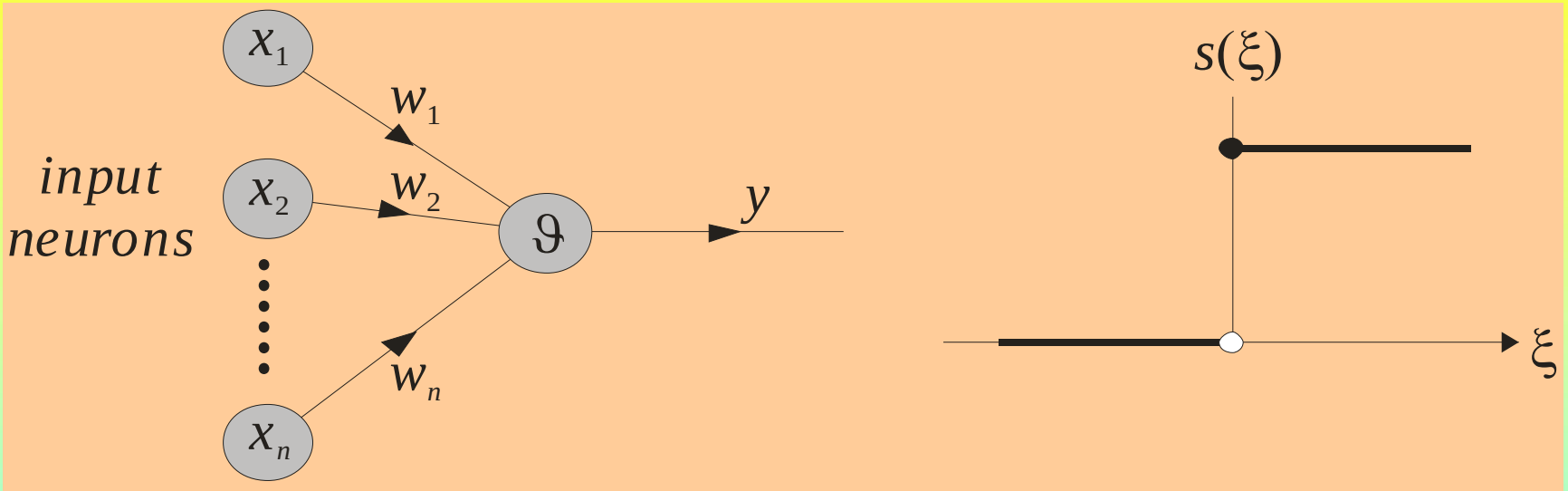


$$\boldsymbol{i} \in \{0,1\}^a, \qquad \boldsymbol{x} \in \{0,1\}^n \qquad \boldsymbol{o} \in \{0,1\}^b,$$



Warren McCulloch

Walter Pitts (1943)



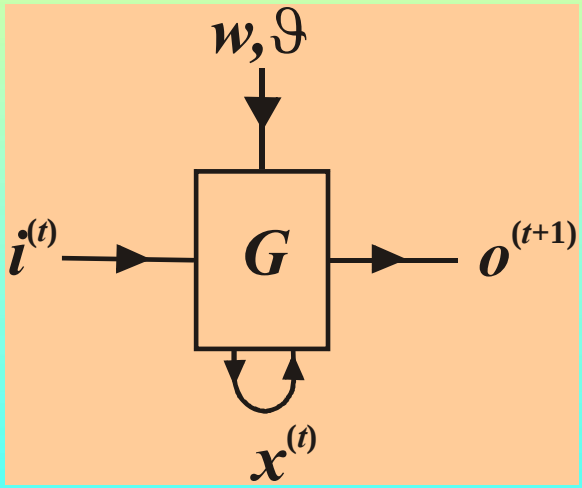
$$y = s(\xi) = s\left(\sum_{i=1}^{p} w_i x_i + \vartheta\right)$$

$$s(\xi) = \begin{cases} 1 & (\text{if } \xi \geq 0) \\ 0 & (\text{otherwise}) \end{cases}$$

Neural network may be expressed as a parametric mapping

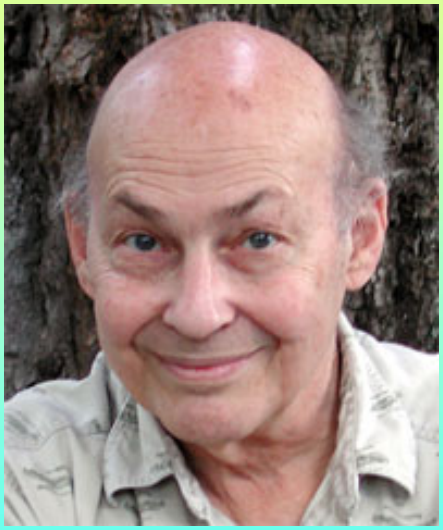
$$\boldsymbol{o}^{(t+1)} = G\left(\boldsymbol{i}^{(t)}; \boldsymbol{x}^{(t)}; \boldsymbol{w}, \vartheta\right)$$

$$G\left(\boldsymbol{w}, \vartheta\right): \{0,1\}^{a} \rightarrow \{0,1\}^{b}$$

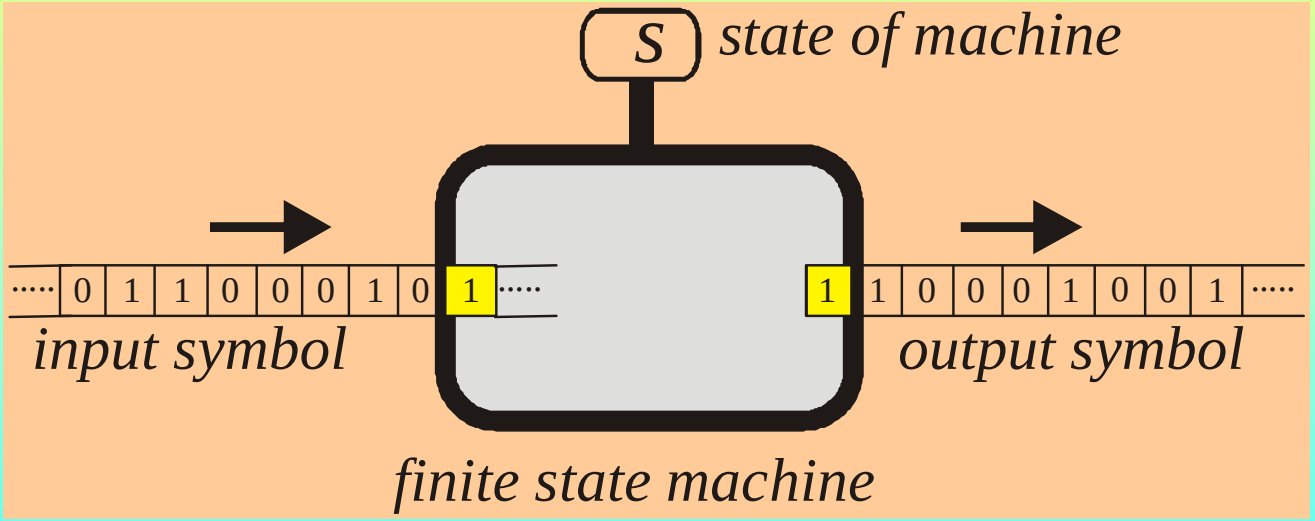


Activities of hidden neurons are necessary as intermediate results for the calculation of activities of output neurons.

3. Finite state machine



Marvin Minsky, 1956)



Finite state machine Works in discrete time events 1, 2,...,t , t+1,... . It contains two tapes: of input symbols and of output symbols, whereas new states are determined by input symbols and an actual state of machine.

$$state_{t+1} = f\left(state_t, input\ symbol_t\right)$$

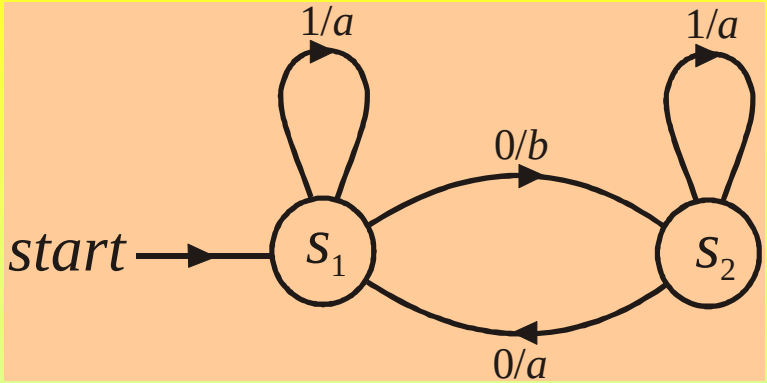
$$output\ symbol_{t+1} = g\left(state_t, input\ symbol_t\right)$$

where function f and g specified given finite state machine and are understand as its basic specification:

(1) ***transition function*** f determines forthcoming state from an actual state and an input symbol,

(2) ***output function*** g determines output symbol from an actual state and an input symbol.

Definícia 2.2. Finite state machine (with input, alternatively called the ***Mealy automat***) is defined as an ordered sextuple $M = (S, I, O, f, g, s_{ini})$, where $S = \{s_1, ..., s_m\}$ is finite set of states, $I = \{i_1, i_2, ..., i_n\}$ is finite set of input symbols, $O = \{o_1, o_2, ..., o_p\}$ is finite set of output symbols, $f : S \times I \to S$ is a transition function, $g : S \times I \to O$ is an input function, and $s_{ini} \in S$ is an initial state.



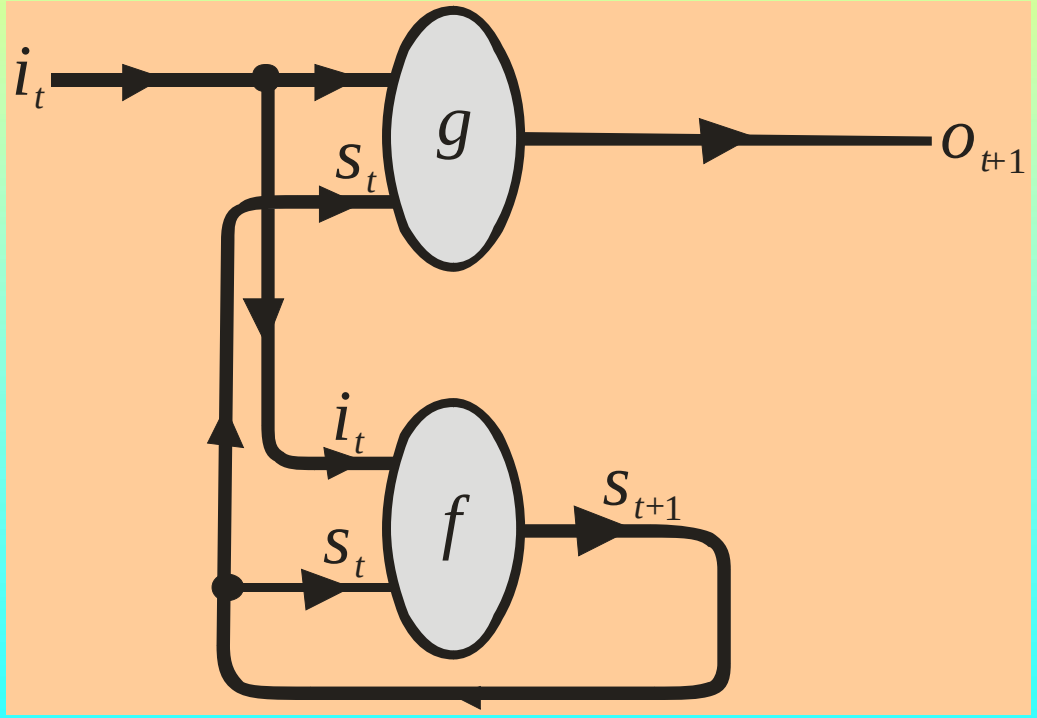
An example of finite state machine, which is composed of two states, $S = \{s_1, s_2\}$, two input symbols, $I = \{0,1\}$, two output symbols, $O = \{a,b\}$, and an initial state is s_1. Transition and output functions are specified in the following table

	f		*g*	
state	*transition function*		*output function*	
	0	1	0	1
s_1	s_2	s_1	*b*	*a*
s_2	s_1	s_2	*a*	*a*

Representation of finite state machine as a calculating devise

$$s^{(t+1)} = f\left(s^{(t)}, i^{(t)}\right)$$

$$o^{(t+1)} = g\left(s^{(t)}, i^{(t)}\right)$$



Theorem 1. Any neural network may be represented by equivalent finite-state machine with output.

A proof of this theorem will be done by a constructive manner; we demonstrate a simple way how to construct single components from the definition $M = (S, I, O, f, g, s_{ini})$ of finite-state machine:

(1) A set S is composed by all possible binary vectors $\boldsymbol{x}_H$, $S = \{\boldsymbol{x}_H\}$. Let neural network is composed of n_H hidden neurons, then a cardinality (number of states) of the set S is 2^{n_H}.

(2) A set of output symbols is composed of all possible binary vectors $\boldsymbol{x}_I$, $I = \{\boldsymbol{x}_I\}$, a cardinality of this set is 2^{n_I}, where n_I is a number of input neurons.

(3) A set of output symbols is composed of all possible binary vectors $\boldsymbol{x}_O$, $O = \{\boldsymbol{x}_O\}$, a cardinality of this set is 2^{n_O}.

(4) A function $f : S \times I \to S$ assigns to each actual state and an actual output symbol new forthcoming state. This function is specified by a mapping, which is determined by the given neural network

$$\boldsymbol{x}_H^{(t+1)} = F\left(\boldsymbol{x}_I^{(t)} \oplus \boldsymbol{x}_H^{(t)}; \mathcal{N}\right)$$

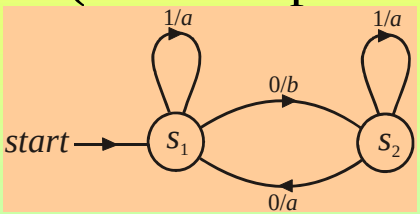
(5) A function $g : S \times I \to O$ assigns to each actual state and an actual output symbol new forthcoming output symbol. In a similar way as for the previous function, this function is also specified by a mapping

$$\boldsymbol{x}_O^{(t+1)} = \tilde{F}\left(\boldsymbol{x}_I^{(t)} \oplus \boldsymbol{x}_H^{(t)}; \mathcal{N}\right)$$

(6) An initial state s_{ini} is usually selected in such a way that all activities of hidden neurons are vanished.

Theorem 2. Any finite-state machine with output (Mealy automaton) may be represented by equivalent recurrent neural network.

A simple demonstration of this theorem will be done by an example of finite-state machine with state diagram (see transparency 14)



This machine is specified by a transfer and output function *f* and *g*, which can be expressed as a Boolean function specified by the following two tables:

(1) Transfer function $state_{t+1} = f\left(state_t, input\ symbol_t\right)$:

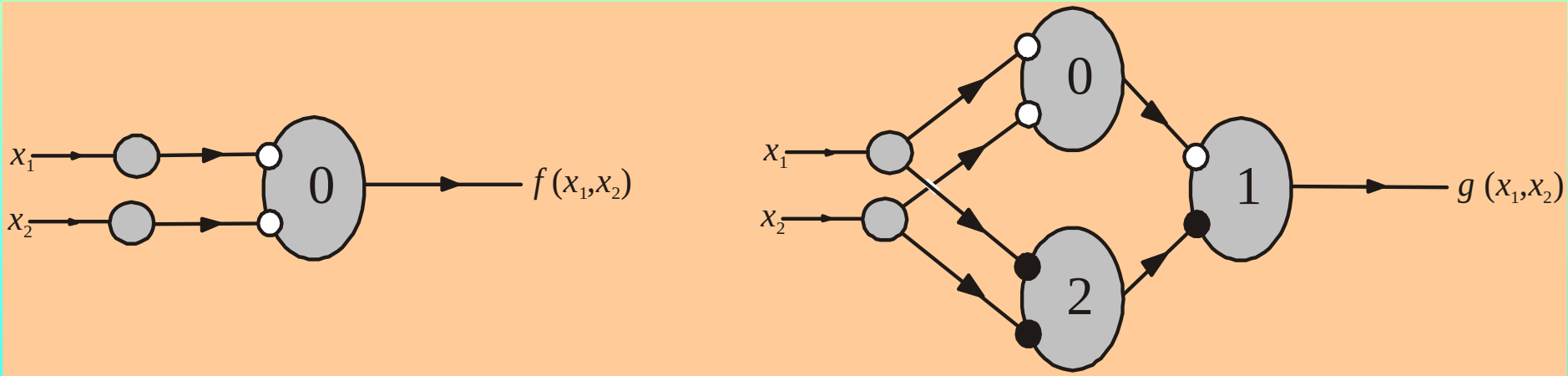
state, input symbol	*transfer function f*
$(s_1,0) \rightarrow (0,0)$	$(b) \rightarrow (1)$
$(s_1,1) \rightarrow (0,1)$	$(a) \rightarrow (0)$
$(s_2,0) \rightarrow (1,0)$	$(a) \rightarrow (0)$
$(s_2,1) \rightarrow (1.1)$	$(a) \rightarrow (0)$

$$f\left(x_1, x_2\right) = \neg x_1 \wedge \neg x_2$$

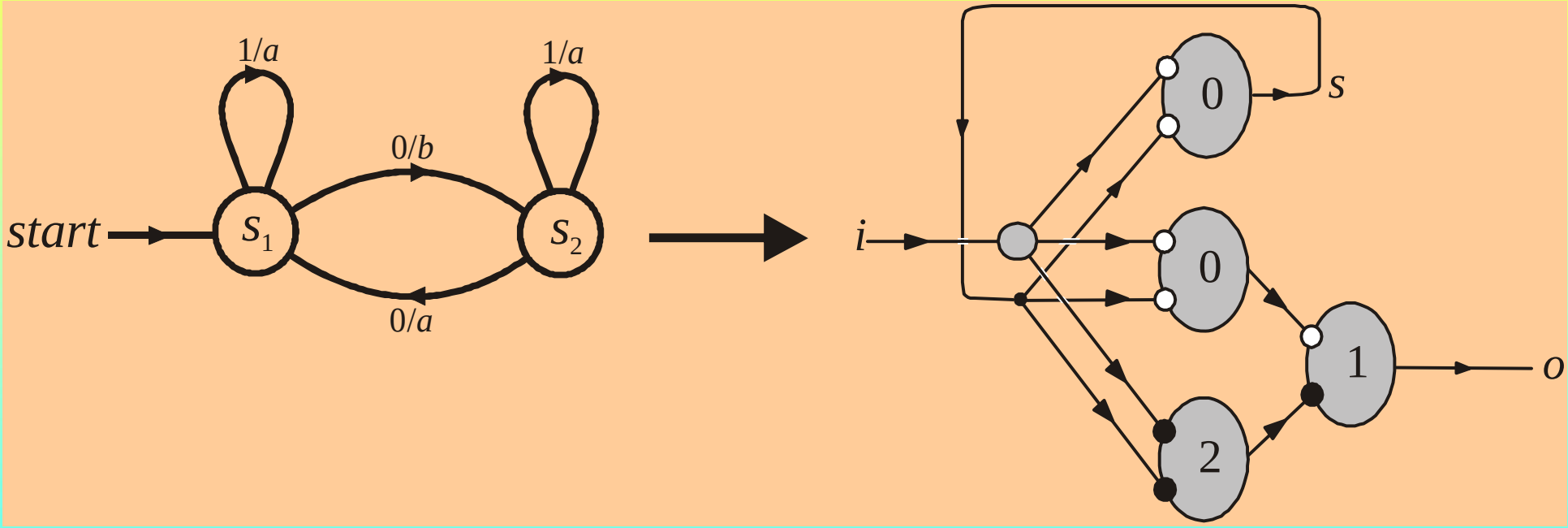
(2) Output function $output\ symbol_{t+1} = g\left(state_t, input\ symbol_t\right)$:

state, output symbol	*output function g*
$(s_1,0) \rightarrow (0,0)$	$(s_2) \rightarrow (1)$
$(s_1,1) \rightarrow (0,1)$	$(s_1) \rightarrow (0)$
$(s_2,0) \rightarrow (1,0)$	$(s_1) \rightarrow (0)$
$(s_2,1) \rightarrow (1,1)$	$(s_2) \rightarrow (1)$

$$g\left(x_1, x_2\right) = \left(\neg x_1 \wedge \neg x_2\right) \vee \left(x_1 \wedge x_2\right)$$



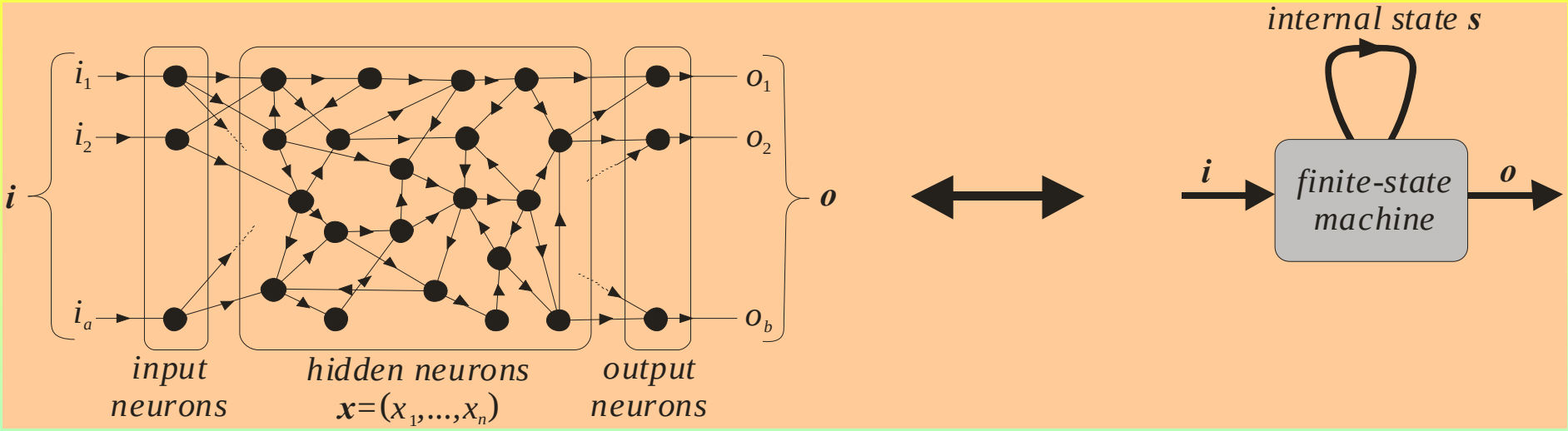
Recurrent neural network, which represents a finite-state machine



Theorems 1 and 2 make possible to study a classical problem of connectionism (neural networks), a relationship between a ***subsymbolic representation*** (neural, which is represented by patterns composed of neural activities) and a ***symbolic representation*** (which is used by classical AI):

(1) According to the Theorem 1, each subsymbolic neural network can be transformed onto symbolic finite-state machine, whereas symbols may be created by making natural numbers that are assigned to binary vectors of activities.

(2) According to the Theorem 2, each symbolic finite-state machine may be transformed onto an equivalent subsymbolic neural network. Symbols from the representation of finite-state machine are represented by binary vectors.



The mentioned theory for the transition from subsymbolic to symbolic representation (and conversely) may be used as a theoretical bases for a study of relationships between symbolic and subsymbolic approaches.