

+

+

On Computational Limitations of Neural Network Architectures

Achim Hoffmann

+

1

In short

A powerful method for analyzing the computational abilities of neural networks based on algorithmic information theory is introduced.

It is shown that the idea of many interacting computing units does not essentially facilitate the task of constructing intelligent systems.

Furthermore, it is shown that the same holds for building powerful learning systems. This holds independently from the epistemological problems of inductive inference.

Overview

- Describing neural networks
- Algorithmic information theory
- The complexity measure for neural networks
- Computational Limits of a particular net structure
- Limitations of learning in neural networks
- Conclusions

Describing neural networks

In general the following two aspects can be distinguished.

a) the functionality of a single neuron.

Often a certain threshold function of the sum of the weighted inputs to the neuron is proposed.

b) the topological organization of a complete network consisting of a large number of neurons.

Often nets are organized in layers. Thus, nets can be distinguished depending on their number of layers.

Describing neural networks

Each node ν in a neural network can be described by the following items:

- The number i of input signals of the particular node
- The nodes in the network whose output signals are connected to each input of ν
- The specification of the I/O behavior of ν .

Describing neural networks

The specification of the I/O behavior of ν

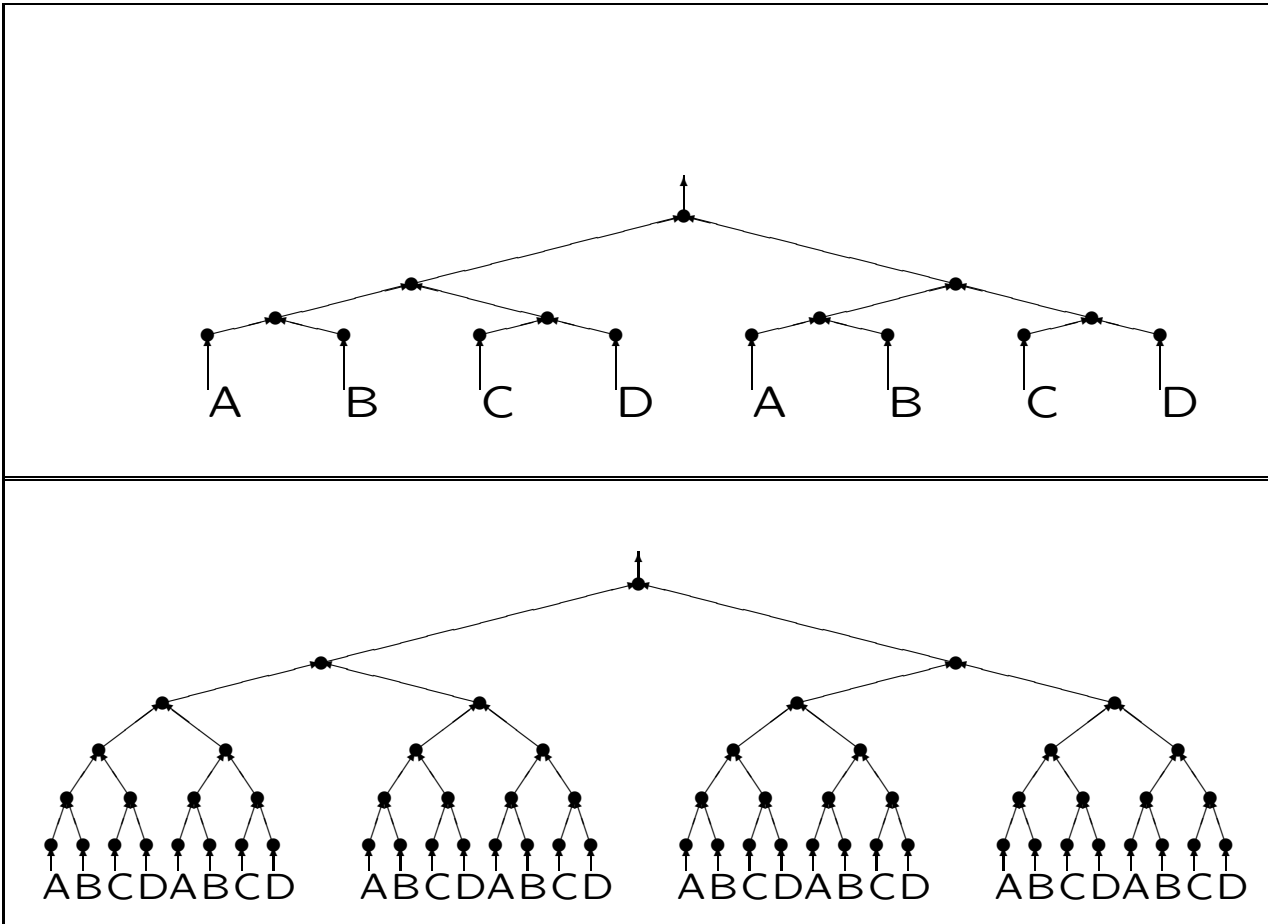
ν may be in different internal states. Let the set of all possible internal states be S_ν . For each computation step of the network, ν computes a function

$f : \{0, 1\}^i \times S_\nu \rightarrow \{0, 1\}$ as output value of ν . Furthermore, ν possibly changes its internal state determined by a function

$g : \{0, 1\}^i \times S_\nu \rightarrow S_\nu$. Both functions f and g are encoded as programs p_f, p_g of minimal length.

+

+



Two neural networks with a similar structure.

+

7

Algorithmic Information Theory

- The amount of information necessary for printing certain strings is measured.
- Only **binary** strings consisting of '0's and '1's are considered.
- The length of the shortest program for printing a certain string s is called its Kolmogorov complexity $K(s)$.

+

+

Examples

Strings of **small** Kolmogorov complexity:

1111111111111111 or

00000000000000000000 or

1010101010101010 etc.

Strings of rather **large** Kolmogorov complexity:

1000100111011001011101010 or

1001111010010110110111001 etc.

+

The complexity of a neural net N

Definition Let $descr(N)$ be the binary encoded description of an arbitrary discrete neural net N . Then, the complexity of N **comp(N)** is given by the Kolmogorov complexity of $descr(N)$

$$comp(N) = K(descr(N))$$

Note: $comp(N)$ reflects the minimal amount of engineering work necessary for designing the network N .

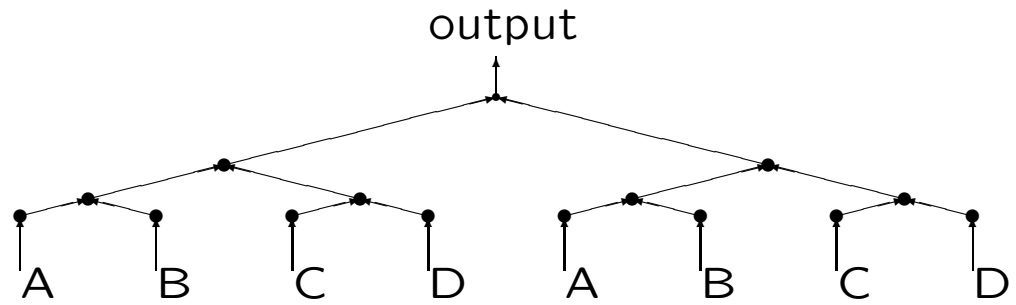
Computational Limitations

Definition Let N be a static discrete neural network with i binary input signals s_1, \dots, s_i and one binary output signal. Then the output behavior of N **is in accordance to a binary string** s of length 2^i , **iff** for any binary number b of the i digits applied as binary input values to N , N outputs exactly the value at the b^{th} position in s .

Theorem Let N be an arbitrary static discrete neural network. Then, N 's output behavior must be in accordance to some binary sequence s with a Kolmogorov complexity $K(s) \leq comp(N) + \text{const}$ for a small constant const .

+

+



A	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
B	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
C	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
D	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

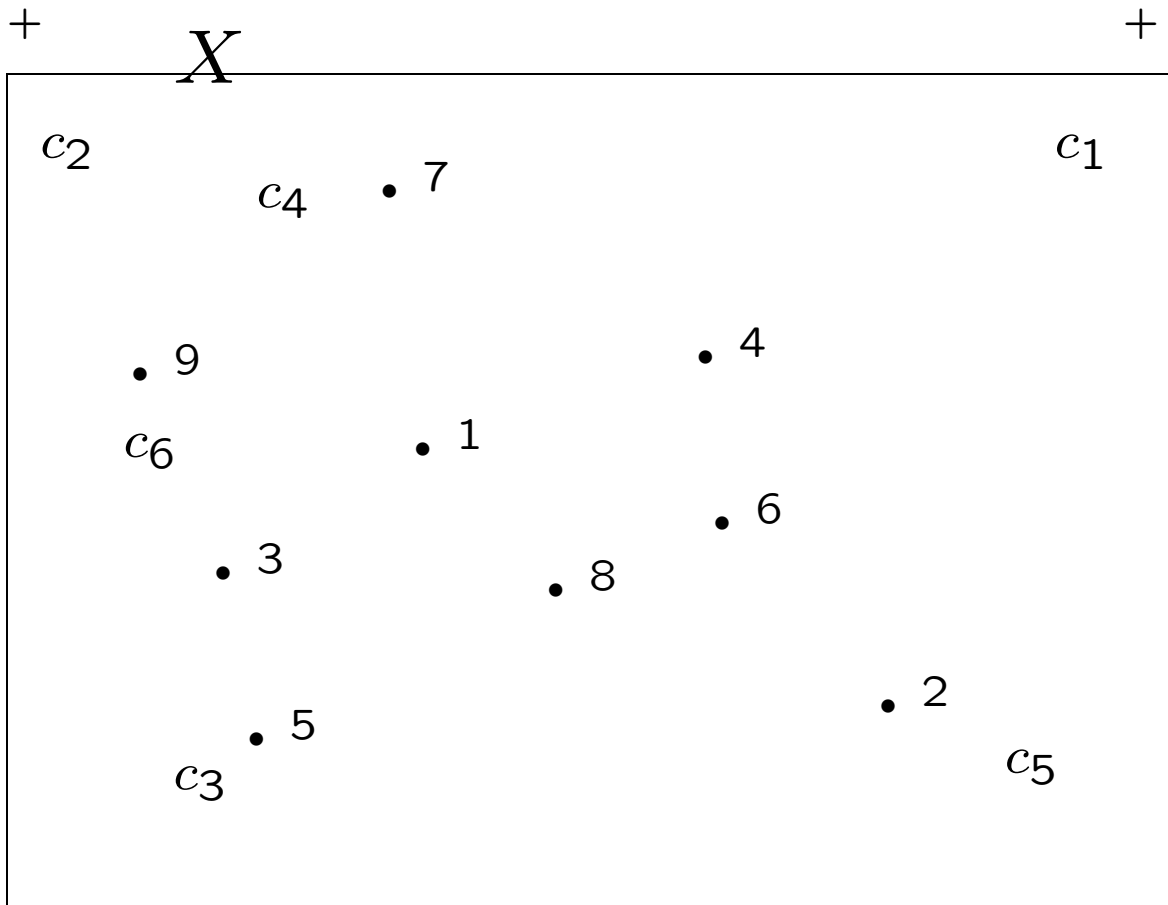
S 0 0 1 0 0 0 1 1 0 1 1 0 1 1 1 0

+

12

Learning in Neural Networks

- We consider a set of objects X .
- The learning task: Determining for each object in X whether it belongs to the class to learn or not.
- A concept is a subset of X . A concept class C is a set of concepts (subsets) of X .
- For any learning system L there is exactly one concept class $C \subseteq 2^X$ that underlies L .



$$X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}, C = \{c_1, c_2, c_3, c_4, c_5, c_6\}.$$

$$c_1 = \{1, 2, 3, 4, 5, 6, 7, 8, 9\},$$

$$c_2 = \{\},$$

$$c_3 = \{1, 3, 5, 7, 9\},$$

$$c_4 = \{1, 4, 6, 7\},$$

$$c_5 = \{2, 4, 6, 8\},$$

$$c_6 = \{1, 2, 4, 6, 9\}$$

+

The binary string representation $s(c)$ of a concept $c \subseteq X$ indicates for each object whether it belongs to c by a corresponding '1'.

Definition The complexity $K_{max}(C)$ of a concept class C is given by the Kolmogorov complexity of the most complex concept in C , i.e.

$$K_{max}(C) = \max_{c \in C} [K(s(c))]$$

+

+

Example $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$

$C = \{c_1, c_2, c_3, c_4, c_5, c_6\}$

$c_1 = \{1, 2, 3, 4, 5, 6, 7, 8, 9\};$

$s(c_1) = '111111111'$

$c_2 = \{ \};$

$s(c_2) = '000000000'$

$c_3 = \{1, 3, 5, 7, 9\};$

$s(c_3) = '101010101'$

$c_4 = \{1, 4, 6, 7\};$

$s(c_4) = '100101100'$

$c_5 = \{2, 4, 6, 8\};$

$s(c_5) = '010101010'$

$c_6 = \{1, 2, 4, 6, 9\};$

$s(c_6) = '110101001'$

$K_{max}(C) = K(s(c_6)) = K(110101001)$

+

16

Learning complex concepts

Theorem Let N be a neural net and $comp(N)$ its complexity. Let C be the concept class underlying N . Then there are at least

$$2^{K_{max}(C) - comp(N) - \text{const}}$$

concepts in C , where const is a small constant integer.

Probably approximately correct learning

Assumptions

- Each $x \in X$ appears with a fixed probability according to some probability distribution D on X .
- This holds during the learning phase as well as for the classification phase.

Goals

- Achieving a high probability for correct classification
- Achieving the above goal with a high confidence probability

Probably approximately correct learning

Definition Let C be a concept class. We say a learning system L **pac-learns** C iff

$$(\forall c_t \in C)(\forall D)(\forall \varepsilon > 0)(\forall \delta > 0)$$

L classifies correctly an object x randomly chosen according to D with probability at least of $1 - \varepsilon$. This has to happen with a confidence probability of at least $1 - \delta$.

Probably approximately correct learning

Theorem Let N be a neural network. Let C be the concept class underlying N .

Let be $0 < \varepsilon \leq \frac{1}{4}$; $0 < \delta \leq \frac{1}{100}$. Then for pac -learning C , N requires at least

$$\frac{K_{max}(C) - \text{comp}(N) - \text{const}}{32\varepsilon \log_2 |X|}$$

examples randomly chosen according to D .
where const is a small constant integer.

Conclusions

- The potential of neural networks for modeling intelligent behavior is essentially limited by the complexity of their architectures.
- The ability of systems to behave intelligently as well as to learn does not increase by simply using many interacting computing units.
- Instead the topology of the network has to be rather irregular !

Conclusions

- With any approach, intelligent neural network architectures require much engineering work.
- *Simple principles* cannot embody the essential features necessary for building intelligent systems.
- Any potential advantage of neural nets for cognitive modelling will become more and more neglectable with an increasing complexity of the system.