#54

BICA VPS 2023

Method of logical interpretation of neural network solutions.

L.A. Lyutikova[1]. [1]Institute of Applied Mathematics and Automation KBSC RAS (IAMA KBSC RAS), Nalchik, Russia,
lylarisa@yandex.ru

IAMA KBSC RAS
INSTITUTE OF APPLIED MATHEMATICS AND AUTOMATION OF KABARDIN-BALKAR SCIENTIFIC CENTER OF RAS

SUMMARY

This paper proposes a method for logical interpretation of neural network solutions Using Boolean integral-differential calculus, it considers possible logical relationships between the input data and the results of the decisions.

INTRODUCTION

Neural networks are complex mathematical models that can find complex and non-obvious patterns in data. However, understanding how exactly these patterns were found is often not clear due to the fact that neural networks usually operate in a non-linear space.

In order to logically interpret the operation of a neural network, various methods can be used that help visualize and analyze the internal processes occurring in the network.

We will try to establish the logical patterns that have arisen in a particular trained neural network without taking into account its structure and the value of the weights of this neural network. This will be an interpretation similar to the comparison model. A set of logical functions will act as such a model. The input and output data will be the values at the input of the neural network and at the output corresponding to these data.

.

APPROACH

Then the mathematical formulation of the problem has the following form.
Let $X = \{x_1, x_2, \dots, x_n\}$, $x_i \in \{0,1,\dots,k_i - 1\}$, where $k_i \in [2,\dots,N]$, $N \in Z$, is the set of neural network inputs. $Y = \{y_1, y_2, \dots, y_m\}$ – many exits, each exit y_i the result of processing specific input values by the neural network
$x_1(y_i),\dots,x_n(y_i): y_i = f(x_1(y_i),\dots,x_n(y_i))$.

$$\begin{pmatrix} x_1(y_1) & x_2(y_1) & \dots & x_n(y_1) \\ x_1(y_2) & x_2(y_2) & \dots & x_n(y_2) \\ \dots & \dots & \dots & \dots \\ x_1(y_m) & x_2(y_m) & \dots & x_n(y_m) \end{pmatrix} \to \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_m \end{pmatrix}.$$

METHODS

•A logical function that reflects the relationship between specific input and output values of a neural network can be found by solving the following equation:

- $\frac{\partial f_i}{\partial P(y_i)} = x_1 \& x_2 \& \dots x_n$

•Then, based on the definition of the Boolean integral, we will have four functions as a solution:

- $f_{1i} = x_{i1} \& x_{i2} \dots \& x_{in} \& P(y_i)$
- $f_{2i} = \overline{x_{i1} \& x_{i2} \dots \& x_{in} \& P(y_i)}$
- $f_{3i} = x_{i1} \& x_{i2} \dots \& x_{in} \to P(y_i)$
- $f_{4i} = \overline{x_{i1} \& x_{i2} \dots \& x_{in} \to P(y_i)}$

RESULTS

Since the outputs in a neural network are independent, it is logical to imagine that the function that combines the functions of each input is either their conjunction or disjunction.
Thus, we have two variants of functions that interpret the dependence of input and output data for each case. And two options for functions that can combine all these solutions.
If we consider as initial functions at each given input and output $f_{1i} = x_{i1} \& x_{i2} \dots \& x_{in} \& P(y_i)$ that

$$\bigvee_{i=1}^{m} f_i = x_{i1} \& x_{i2} \dots \& x_{in} \& P(y_i) = 0\,.$$

If we consider the conjunction as the initial functions at each given input and output, and the disjunction as the unifying function

$$\bigvee_{i=1}^{m} f_{1i} = x_{i1} \& x_{i2} \dots \& x_{in} \& P(y_i)\,,$$
$$\bigvee f_i = x_{i1} \& x_{i2} \dots \& x_{in} \& P(y_i)\,,$$

then it will be a neural network capable of giving only those answers that we have considered. Logically, this is a function of

$$F(x_1,\dots,x_n, P^{\sigma}(y_1),\dots,P^{\sigma}(y_m)) = \begin{cases} 1 & if \quad x_{i1} \& x_{i2} \dots \& x_{in} \& P(y_i) \\ 0 \end{cases}$$

This option does not give the opportunity for conclusions, when at least some values will differ from the given ones.

.

ANALYSIS

If we consider the implication as the initial functions at each given input and output, and the conjunction as the union, we obtain the function

$$f(X) = \bigvee_{j=1}^{m} \left(\bigvee_{i=1}^{n} \overline{x_i} \to P(y_j) \right)$$

This function has a number of interesting properties.

DISCUSSION

Example: suppose we have two inputs and two outputs. One input is the values (0,1) at the output of the object "a", the second input is the values (1,1) at the output of the object "b". Let's build functions that reflect the relationship between (0,1) and "a". Initial For our example data:

$$f_i = \overline{x_1} \& x_2 \to P(a) = x_1 \vee \overline{x_2} \vee P(a)$$
$$f_{3i} = x_1 \& x_2 \to P(b) = \overline{x_1} \vee \overline{x_2} \vee P(b)$$
$$(x_1 \vee \overline{x_2} \vee P(a))\overline{(x_1 \vee \overline{x_2} \vee P(b))} =$$
$$= \overline{x_2} \vee \overline{x_1}\, P(a) \vee x_1\, P(b) \vee P(a)\, P(b)$$

CONCLUSIONS

Because of the considered method, it can be argued that for the logical interpretation of a correctly functioning neural network, it is possible to construct a function that will give an idea of the hidden patterns, the existing classes, and the most important features in the processed data. This approach does not take into account the weight, structure, method of learning, rather, it refers to the interpretation through comparison models, and gives a complete picture of the properties of the data under study on the considered set of solutions