УДК 004.085
DOI: 10.26102/2310-6018/2023.43.4.037

Использование логических методов для анализа решений нейронной сети

Л.А. Лютикова✉

Институт прикладной математики и автоматизации КБНЦ РАН,
Нальчик, Российская Федерация

Резюме. В данной работе предлагается метод интерпретации решений нейронных сетей, основанный на использовании булевого интегро-дифференциального исчисления. Этот метод позволяет исследовать логику принятия решений нейронными сетями и определить наиболее важные признаки, влияющие на их решения. Метод может быть применен для задач классификации, особенно в случаях, когда каждый признак может быть представлен как k-значная переменная. В работе рассматриваются локальная и глобальная интерпретации решений. На первом этапе происходит связывание каждого входного вектора с соответствующим выходом нейронной сети. Затем путем решения булевого уравнения находятся логические функции, которые адекватно отражают входные данные и соответствующие им выходы. На втором этапе глобальной интерпретации строятся функции, объединяющие ранее найденные логические функции. Этот выбор функций основывается на их способности наиболее точно отражать решения нейронной сети и исследуемой области. Полученная функция обладает интерпретируемостью, модифицируемостью и способностью представлять полное множество решений, соответствующих заданному запросу. Она также выделяет наиболее значимые признаки для каждого решения. В работе рассматривается практическая реализация метода на примере нейронной сети, обученной на основе структуры и входных данных, состоящих из ответов на анкетные вопросы, с выходным узлом, предсказывающим диагноз. Параллельно с разработкой нейронной сети строится интерпретационная модель, которая позволяет выявить наиболее важные признаки для каждого диагноза на основе решений нейронной сети. Кроме того, в случаях с пограничными решениями, когда нейронная сеть предоставляет только одно возможное решение, интерпретационная модель способна найти все возможные решения с заранее заданной точностью, что помогает избежать ошибок в принятии решений.

Ключевые слова: нейронные сети, интерпретатор, связи, булево дифференцирование, входные данные, анализ, скрытые закономерности.

Для цитирования: Лютикова Л.А. Использование логических методов для анализа решений нейронной сети. *Моделирование, оптимизация и информационные технологии.* 2023;11(4). URL: https://moitvivt.ru/ru/journal/pdf?id=1477 DOI: 10.26102/2310-6018/2023.43.4.037

A method for constructing a logical model for interpreting the decisions of a trained neural network

L.A. Lyutikova✉

Institute of Applied Mathematics and Automation KBSC RAS,
Nalchik, the Russian Federation

Abstract. In this paper, we propose a method for interpreting neural network solutions based on the use of Boolean integro-differential calculus. This method allows you to investigate the logic of decision-making by neural networks and determine the most important signs that affect their decisions. The method can be applied to classification problems, especially in cases where each feature can be represented as a k-valued variable. The paper considers local and global interpretations of solutions. At the first stage, each input vector is associated with the corresponding output of the neural network. Then,

by solving a Boolean equation, logical functions are found that adequately reflect the input data and their corresponding outputs. At the second stage, global interpretation, functions are constructed that combine previously found logical functions. This choice of functions is based on their ability to most accurately reflect the decisions of the neural network and the study area. At the second stage, global interpretation, functions are constructed that combine previously found logical functions. This choice of functions is based on their ability to most accurately reflect the decisions of the neural network and the study area. The resulting function has interpretability, modifiability and the ability to represent a complete set of solutions corresponding to a given query. It also highlights the most significant features for each solution. The paper considers the practical implementation of the method on the example of a neural network trained on the basis of the structure and input data consisting of answers to questionnaire questions, with an output node predicting the diagnosis. In parallel with the development of the neural network, an interpretive model is being built, which allows identifying the most important signs for each diagnosis based on the decisions of the neural network. In addition, in cases with boundary solutions, when the neural network provides only one possible solution, the interpretative model is able to find all possible solutions with a predetermined accuracy, which helps to avoid mistakes in decision-making.

Keywords: neural networks, interpreter, connections, Boolean differentiation, input data, analysis, hidden patterns.

For citation: Lyutikova L.A. A method for constructing a logical model for interpreting the decisions of a trained neural network. *Modeling, Optimization and Information Technology*. 2023;11(4). URL: https://moitvivt.ru/ru/journal/pdf?id=1477 DOI: 10.26102/2310-6018/2023.43.4.037 (In Russ.).

Введение

Нейронные сети – это мощный инструмент машинного обучения, который применяется для решения разнообразных задач, включая классификацию, регрессию, генерацию текста, распознавание образов и многое другое. Они обладают способностью обрабатывать сложные данные и выявлять сложные паттерны, что делает их очень эффективными во многих областях.

Однако одна из основных проблем с нейронными сетями заключается в их интерпретируемости. Интерпретируемость в контексте машинного обучения означает способность объяснить, как модель принимает свои решения. Например, для линейной регрессии или деревьев решений легко понять, какие факторы влияют на их выводы. Однако с нейронными сетями это гораздо сложнее из-за их сложной структуры и нелинейных функций активации.

Существует несколько подходов, которые помогают в интерпретации решений нейронных сетей. Один из них – это использование методов анализа важности признаков, которые позволяют определить, какие признаки вносят наибольший вклад в принимаемые решения. Например, можно использовать градиентные методы, чтобы выявить, как изменение входных признаков влияет на изменение выхода сети. Также существуют методы, основанные на анализе активаций и весов нейронов, которые помогают понять, какие части сети отвечают за определенные аспекты задачи.

Однако стоит отметить, что все эти методы дают только приближенное понимание работы модели. Нейронные сети остаются в значительной степени "черными ящиками", и полное понимание их решений может быть сложным или невозможным. Интерпретируемость нейронных сетей остается активной областью исследований [1-4].

В данной работе рассматривается метод логической интерпретации решений нейронной сети.

Существуют различные логические подходы к интерпретации нейронных сетей. *Выделение логических правил из нейронной сети, которые объясняют принимаемые ее решения.* Это может быть сделано, например, путем анализа активаций нейронов и определения, какие комбинации активаций соответствуют определенным логическим

условиям. Таким образом, можно получить набор правил, которые объясняют, какие факторы влияют на решения сети.

Можно применить подход, при котором нейронная сеть интерпретируется как логическая формула, которая описывает ее поведение. Это может быть сделано, например, путем представления входов, выходов и весов сети в виде логических переменных и использования логических операций для описания их взаимодействия. Таким образом, можно получить логическое выражение, которое представляет связь между входами и выходами сети.

Еще один подход заключается в анализе решений, принимаемых нейронной сетью, и выявлении логических закономерностей в этих решениях. Например, можно исследовать, какие входные признаки и их комбинации приводят к определенным выходным классам или значениям. Таким образом, можно вывести логические правила, которые объясняют, какие факторы влияют на решения сети.

Каждый из этих подходов имеет свои преимущества и недостатки, и выбор конкретного метода зависит от задачи и требований интерпретации. Однако важно отметить, что интерпретация нейронных сетей остается сложной проблемой, и разработка эффективных методов интерпретации является активной областью исследований [2, 5].

В работе предлагается метод, основанный на булевом интегро-дифференциальном исчислении. Этот метод позволяет выявить логические закономерности, которые возникают в обученной нейронной сети, не привязываясь к ее структуре и весам. Он позволяет объяснить работу моделей машинного обучения без необходимости знания их внутренней структуры или алгоритмов обучения [6].

Материалы и методы

Как уже упоминалось, основная цель интерпретации решений нейронной сети заключается в понимании ключевых признаков входных данных, которые имеют наибольшую важность для принятия решений сетью, а также выявлении закономерностей, которые сеть обнаруживает в данных.

Интерпретация позволяет получить ценную информацию о работе сети и ее потенциальных улучшениях, а также понять сущность самих данных.

Предлагаемый метод может быть применен для задач классификации в случаях, когда входные и выходные данные могут быть перекодированы в дискретные значения. Это означает преобразование непрерывных или категориальных признаков в дискретные значения, чтобы метод мог быть применен.

В таком случае каждый признак представим в виде булевой переменной.

Рассматриваемая задача в математической постановке будет иметь следующий вид.

Пусть $X = \{x_1, x_2, \dots, x_n\}$, $x_i \in \{0,1\}$ – множество входов нейронной сети. $Y = \{y_1, y_2, \dots, y_m\}$ – множество выходов, каждый выход y_i – результат обработки нейронной сетью конкретных значений на входе.

$x_1(y_i), \dots, x_n(y_i)$: $x_i(y_j) = 1$, когда значения входа x_i при выходе y_j совпадает со значением этого входа нейронной сети, и $x_i(y_j) = 0$, когда значение на данном входе не совпадают. $y_i = q(x_1(y_i), \dots, x_n(y_i))$,

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

Для решения поставленной задачи рассмотрим метод интегро-дифференциального исчисления булевых функций. Этот подход позволяет построить логическую модель системы, описывающую зависимость выходных значений от входных параметров, и использовать ее для анализа важности каждого из параметров. Введем некоторые понятия и определения.

Чтобы вычислить булеву производную, которая является аналогом классической производной в дифференциальном исчислении, можно использовать различные методы. Один из таких методов – метод конечных разностей. Он основан на вычислении разности между значениями функции при небольших изменениях входных параметров.

Для более точных результатов можно применить другие методы булевой дифференциации, такие как методы символьной дифференциации или методы аппроксимации с использованием логических операций [7].

В целом использование интегрально-дифференциального аппарата булевых функций позволяет провести анализ важности входных параметров в нейронных сетях и получить информацию о влиянии каждого параметра на выходные значения. Это может быть полезным для понимания и оптимизации работы нейронных сетей

Определение. Производная первого порядка $\frac{\partial f}{\partial x_i}$ от булевой функции $f(x_1 \dots, x_n)$ по переменной x_i есть сумма по модулю 2 соответствующих остаточных функций:

$$\frac{\partial f}{\partial x_i} = f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \oplus f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n). \quad (1)$$

Определение. Пусть $g = \frac{\partial f}{\partial x_i}$ – производная функции f по переменной x_i, тогда существует функция $\int g dx_i$, называемая булевым интегралом функции g такая, что $\int g dx_i = x_i g \oplus h$, где h является произвольной булевой функцией переменных $x_1, \dots x_{i-1}, x_{i+1}, .., x_n$.

Булев интеграл является математическим инструментом, который позволяет агрегировать информацию с использованием операций булевой алгебры, таких как конъюнкция (логическое И) и дизъюнкция (логическое ИЛИ). Он использует значимости переменных для более точного отражения их вклада в общий результат. Другими словами, булев интеграл позволяет выразить важность каждой переменной при агрегации информации [8].

Предлагаемый метод интерпретации решений нейронной сети строит локальные интерпретации для каждого объекта, а затем объединяет их в глобальную интерпретацию. Такой подход позволяет учитывать особенности каждого отдельного случая и одновременно получить общее представление о важности признаков в рамках всей модели.

Построение локальная интерпретация

На первом этапе данного метода строится функция, которая связывает значения на входе и выходе нейронной сети.

Каждый выход y_iзависит от значений на входе $x_1(y_i), \dots, x_n(y_i)$, и эта зависимость может быть описана булевой функцией $f_i(x_1(y_i), \dots, x_n(y_i), P^\sigma(y_i))$, где

$$\mathrm{P}^\sigma(y_i) = \begin{cases} \overline{P(y_i)} & \text{при} \quad \sigma = 0 \\ P(y_i) & \text{при} \quad \sigma = 1 \end{cases}. \quad (2)$$

Если на входе нейронной сети были конкретные значения $x_1(y_i), \dots, x_n(y_i)$, на выходе значение y_i, то искомая функция на наборе соответствует заданному выходу $f_i(x_1(y_i), \dots, x_n(y_i), P(y_i)) = 1$, когда значения на входе соответствуют, а на выходе значение не соответствует y_i, то $f_i(x_1(y_i), \dots, x_n(y_i), \overline{P(y_i)})) = 0$.

Пример: предположим, что у нас два входа со значениями (0,1) и объект «а» на выходе. Тогда имеем конечное множество булевых функций, описывающих это условие (Таблица 1).

Таблица 1 – Таблица функций, соответствия входа и выхода
Table 1 – Table of functions, matching input and output

x_1	x_2	$P(a)$	f_i
0	0	0	
0	0	1	
0	1	0	0
0	1	1	1
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Для определения значений функции в точках, на которых она неопределённа, учтем следующие условия: искомая функция должна изменить свои значения на наборе $x_1(y_i)\,\&x_2(y_i)\,\&\ldots x_n(y_i), P(y_i)$, и $x_1(y_i)\,\&x_2(y_i)\,\&\ldots x_n(y_i), \overline{P(y_i)}\ (y_i)$.

Это значит, что производная $\frac{\partial f_i}{\partial P(y_i)} = x_1(y_i)\,\&x_2(y_i)\,\&\ldots x_n(y_i)$, поскольку совокупность входных значений нейронной сети и верный результат на выходе интерпретируется функций как «1», а совокупность входных значений нейронной сети и результат на выходе, не соответствующий выходу нейронной сети, интерпретируется функцией как «0».

Таким образом, мы находим функции для каждого входа и подбираем композицию функций с целью получения обобщающей функции, которая связывает все входы и выходы нейронной сети вместе.

$$F\big(x_1(y_i), \ldots, x_n(y_i), P^{\sigma}(y_1), \ldots, P^{\sigma}(y_n)\big). \tag{3}$$

В связи с тем, что значения на выходах в задачах машинного обучения зависят от конкретных значений на входе, функция должна быть определена таким образом, что при рассмотрении только входных данных они должны соответствовать определенному набору конкретных входных значений. Это можно описать как конъюнкцию всех значений на входе $\&_{j=1}^{m} x_j(y_i)$.

Можно сказать, что логическая функция, которая отражает связь между конкретными входными и выходными значениями нейронной сети, может быть получена путем решения следующего уравнения:

$$\frac{\partial f_i}{\partial P(y_i)} = x_1 \& x_2 \& \ldots \& x_n. \tag{4}$$

Рассматривая совокупность переменных $\&_{j=1}^{m} x_j(y_i)$ как одну переменную, поскольку именно набор этих значений дает конкретное значение на выходе, решением у нас будет четыре функции, булевы производные, которые равны $\&_{j=1}^{m} x_j(y_i)$:

$$\begin{aligned} f_{1i} &= \overline{x_{i1} \& x_{i2} \ldots \& x_{in} \& P(y_i)}, \\ f_{2i} &= x_{i1} \& x_{i2} \ldots \& x_{in} \& P(y_i), \\ f_{3i} &= \overline{x_{i1} \& x_{i2} \ldots \& x_{in} \to P(y_i)}, \\ f_{4i} &= x_{i1} \& x_{i2} \ldots \& x_{in} \to P(y_i). \end{aligned} \tag{5}$$

Причем функции f_{2i} и f_{3i} решениями для наших условий являться не могут.

Следовательно, в данной интерпретации, каждый выход нейронной сети связан с соответствующим ему входом или функцией конъюнкции, или импликацией.

Пример: предположим, что у нас два входа и два выхода. Один вход – это значения (0,1) на выходе объект «a», второй вход – значения (1,1), на выходе объект «b».

Построим функции, которые отражают зависимость между (0,1) и «a». Исходные данные приведены в Таблице 2.

Таблица 2 – Функция значения входов
Table 2 – Input value function

x_1	x_2	$\overline{x_1}\&x_2$
0	0	0
0	1	1
1	0	0
1	1	0

Продемонстрируем таблично функции, соответствующие выражению $\int \overline{x}_1\&x_2 dP(a)$ в Таблице 3.

Таблица 3 – Таблица функций, соответствующих значению $\int \overline{x}_1\&x_2 dP(a)$
Table 3 – Table of functions corresponding to the value

x_1	x_2	$P(a)$	f_{1i}	f_{2i}	f_{3i}	f_{4i}
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	0	1	0	1
0	1	1	1	0	1	0
1	0	0	0	1	1	0
1	0	1	0	1	1	0
1	1	0	0	1	1	0
1	1	1	0	1	1	0

Из Таблицы 3 видно, что четыре функции соответствуют

$$\begin{aligned} f_{1i} &= \overline{x}_1\&x_2\&P(a), \\ f_{2i} &= \overline{\overline{x}_1\&x_2\&P(a)}, \\ f_{3i} &= \overline{x_1}\&x_2 \to P(a), \\ f_{4i} &= \overline{\overline{x_1}\&x_2 \to P(a)}. \end{aligned} \quad (6)$$

Функции f_{2i} и f_{4i} решениями быть не могут, так как у них есть точка (0,1,0), наличие входных сигналов и отсутствие выходного, что противоречит условиям, остается только $f_{1i} = \overline{x_1}\&x_2\&P(a)$, и $f_{3i} = \overline{x_1}\&x_2 \to P(a)$.

Рассмотрим теперь условия вход-значения (1,1) на выходе объект «b» (Таблица 4).

Таблица 4 – Функция значения входов
Table 4 – Input value function

x_1	x_2	$\overline{x_1}\&x_2$
0	0	0
0	1	0
1	0	0
1	1	1

Продемонстрируем таблично функции, соответствующие выражению $\int x_1\&x_2 dP(b)$ (Таблица 5).

Таблица 5 – Таблица функций, соответствующих значению $\int \&x_2 dP(b)$
Table 5 – Table of functions corresponding to the value

x_1	x_2	$P(b)$	f_{1i}	f_{2i}	f_{3i}	f_{4i}
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	0	1	1	0
0	1	1	0	1	1	0
1	0	0	0	1	1	0
1	0	1	0	1	1	0
1	1	0	0	1	0	1
1	1	1	1	0	1	0

Удовлетворяющие условия решения

$$f_{1i} = x_1\&x_2\&P(b),$$
$$f_{3i} = x_1\&x_2 \to P(b). \quad (7)$$

Для создания логической функции, которая объединяет функции каждого выхода в суперпозицию, требуется построить композицию этих функций в одну функцию, зависящую от всех входных параметров одновременно [9].

Метод глобальной интерпретации модели нейросетевых решений

На втором этапе для объединения функций каждого выхода в задачах машинного обучения, учитывая, что выходы являются независимыми, нужно использовать функции, обладающие свойством коммутативности. Среди шести булевых функций, обладающих свойством коммутативности, можно исключить константы, функцию эквивалентности, поскольку локальные решения не обязаны быть эквивалентными, сложение по модулю два, поскольку эта функция расстояния. Остается конъюнкция, дизъюнкция и их отрицания. Отрицания в нашем случае можно исключить.

Таким образом, получаем две возможные функции для интерпретации зависимости между входными и выходными данными в каждом локальном случае. И две возможные функции для объединения всех этих решений.

Если в качестве исходных функций на каждом заданном входе и выходе рассматривать $f_{1i} = x_{i1}\&x_{i2}\ldots\&x_{in}\&P(y_i)$, то

$$\overset{i=1}{\underset{m}{\&}} f_i = x_{i1}\&x_{i2}\ldots\&x_{in}\&P(y_i) = 0. \quad (8)$$

Если в качестве исходных функций на каждом заданном входе и выходе рассматривать конъюнкцию, а в качестве объединяющей функции дизъюнкцию

$$\begin{matrix} f_{1i} = x_{i1}\&x_{i2}\ldots\&x_{in}\&P(y_i), \\ \overset{i=1}{\overset{m}{\vee}} f_i = x_{i1}\&x_{i2}\ldots\&x_{in}\&P(y_i), \end{matrix} \quad (9)$$

то это будет нейронная сеть, способная давать только те ответы, которые мы рассматривали. Логически это функция от

$$F\left(x_1,\ldots,x_n,P^{\sigma}(y_1),\ldots,P^{\sigma}(y_m)\right) = \left\{\frac{1 \quad if \quad x_{i1}\&x_{i2}\ldots\&x_{in}\&P(y_i)}{0}. \right. \quad (10)$$

Если мы рассмотрим импликацию в качестве исходных функций для каждого входа и выхода, а дизъюнкцию в качестве операции объединения, то при отличии хотя бы одного значения от заданного получим результат, равный единице.

Другие способы выражения:

1. При использовании импликации в качестве функций для каждого входа и выхода и дизъюнкции в качестве операции объединения, получим положительный результат, если хотя бы одно значение отличается от заданного.

2. Объединение импликаций с помощью дизъюнкции позволяет получить единичный результат, если хотя бы одно из входных и выходных значений не соответствует заданному.

Такие формулировки позволяют описать использование импликации и дизъюнкции для получения единичного результата, когда хотя бы одно значение отличается от заданного.

Для нашего примера:

$$\begin{matrix} f_{3i} = \overline{x_1}\&x_2 \to P(a) = x_1 \vee \overline{x_2} \vee P(a), \\ f_{3i} = x_1\&x_2 \to P(b) = \overline{x_1} \vee \overline{x_2} \vee P(b), \\ x_1 \vee \overline{x_2} \vee P(a) \vee \overline{x_1} \vee \overline{x_2} \vee P(b) = 1. \end{matrix} \quad (11)$$

Если мы рассматриваем импликацию в качестве исходных функций для каждого входа и выхода, а конъюнкцию в качестве операции объединения, то получаем функцию-конъюнкцию, которая будет истинной только в тех случаях, когда все значения входов и выходов соответствуют заданным.

$$f(X) = \overset{j=1}{\overset{m}{\&}} \left(\overset{i=1}{\overset{n}{\&}} \overline{x_i} \to P\left(y_j\right) \right). \quad (12)$$

Эта функция обладает рядом интересных свойств [9].

Для нашего примера:

$$\begin{matrix} f_{3i} = \overline{x_1}\&x_2 \to P(a) = x_1 \vee \overline{x_2} \vee P(a), \\ \overline{f_{3i}} = x_1\&x_2 \to P(b) = \overline{x_1} \vee \overline{x_2} \vee P(b), \\ (x_1 \vee \overline{x_2} \vee P(a))\&\overline{(x_1 \vee \overline{x_2} \vee P(b))} = \overline{x_2} \vee \overline{x_1}\,P(a) \vee x_1\,P(b) \vee P(a)\,P(b). \end{matrix} \quad (13)$$

Мы можем утверждать, что у нас нет решений, содержащих $\overline{x_2}$. Для того чтобы отличить один объект от другого достаточно одной переменной x_1.

Результаты

Практическая реализация предлагаемого метода была рассмотрена на задаче медицинской диагностики. Для разработки алгоритма адекватной диагностики гастрита у остальных пациентов были предоставлены данные о 132 пациентах, которым проводили гастроэнтерологические обследования для диагностики гастрита. В историях

болезни каждого пациента было учтено 28 симптомов заболевания, каждый из которых имел от 2 до 4 вариантов ответов. При выборе этих признаков мы опирались на клиническую практику и включили различные виды обследований.

В результате анализа было выявлено 17 типов гастрита. Для разработки алгоритма диагностики необходимо использовать имеющиеся данные и информацию о симптомах, чтобы классифицировать остальных пациентов и определить их тип гастрита**.** Приведенный ниже Рисунок 1 демонстрирует образец анкеты.

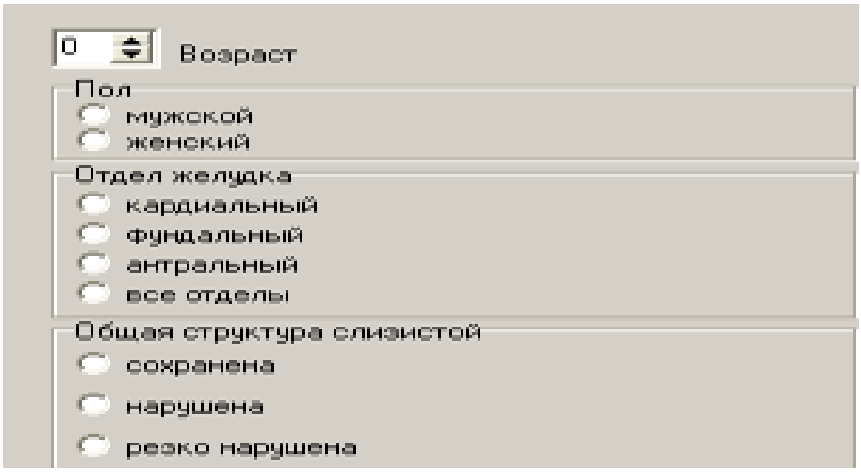

Рисунок 1 – Образец анкеты
Figure 1 – Sample questionnaire

Для решения задачи диагностики гастрита была разработана нейронная сеть с архитектурой, названной Сигма-Пи (Sigma-Pi). Эта архитектура объединяет элементы сигмоидной (sigmoid) и пи (pi) нейронных сетей. В ней используется функция активации сигмоида для нелинейного преобразования входных данных и функция активации пи для объединения информации из различных входных узлов.

Входные данные для нейронной сети состоят из ответов на вопросы анкеты, а выходной узел представляет собой предсказание диагноза гастрита для данного пациента. После обучения на исходных данных нейронная сеть достигла точности предсказания диагноза в 94 %.

Обсуждение

Параллельно с разработкой нейронной сети была построена предложенная интерпретационная модель. Результаты работы модели для произвольно введенного запроса приведены на Рисунке 2.

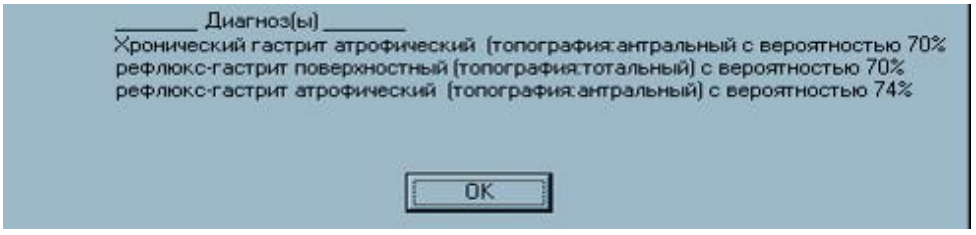

Рисунок 2 – Результат работы модели для произвольно введенного запроса
Figure 2 – Result of the model for an arbitrarily entered query

В результате работы данной модели возможно получить несколько диагнозов, которые могут соответствовать данному запросу. Модель объединяет признаки заболевания в классы на основе анализа данных и обучения**.**

Это позволяет определить, какие признаки и комбинации признаков характерны для каждого класса гастрита. Фрагмент общих признаков для группы гастритов приведен на Нисунке 3.

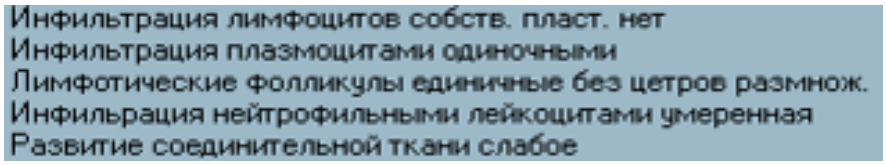

Рисунок 3 – Фрагмент общих признаков для группы гастритов
Figure 3 – Fragment of common features for a group of gastritis

Также возможно выявить группу признаков, которые являются характерными и индивидуальными для конкретного диагноза. Это позволяет получить более глубокое понимание природы исследуемых данных и их различий

Заключение

В сравнение с другими моделями интерпретации предложенный подход обладает рядом достоинств и недостатков. На данном этапе в практическом приложении он не может претендовать на универсальность. При обработке естественного языка более адекватными являются интерпретационные модели аттеншен (attention), визуализация весов аттеншен и другие аналитические инструменты. Данный подход не проводит внутреннее исследование «черного ящика». Для проведения внутреннего исследования «черного ящика» в нейронных сетях можно использовать библиотеку Lucid, разработанную в Google.

Метод применим, когда нам необходимо понять исследуемую область. Выявить скрытые логические закономерности в данных, создать модель, которая способна корректировать работу нейросетевого метода при наличии неточных или зашумленных данных. Кроме того, интерпретационная модель может стать методом машинного обучения, который обладает свойством модифицируемости и возможностью минимизации данных, необходимых для корректного решения [10].

СПИСОК ИСТОЧНИКОВ

1. Шибзухов З.М. Корректные алгоритмы агрегирования операций. *Распознавание образов и анализ изображений*. 2014;24:377–382.
2. Аверкин А.Н., Ярушев С.А. Обзор исследований в области разработки методов извлечения правил из искусственных нейронных сетей. *Известия РАН. Теория и системы управления*. 2021;6:106–121.
3. Ashley I., Naimi Laura B., Balzer Multilevel generalization: an introduction to super learning. *European Journal of Epidemiology*. 2018;33:459–464.
4. Haoxiang W., Smith S. Big data analysis and perturbation using a data mining algorithm. *Journal of Soft Computing Paradigm*. 2021;3-01:19–28.
5. Ribeiro, M. T., Singh, S., Guestrin C. "Why should I trust you?": Explaining the predictions of any classifier. *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 2016:1135-1144.

6. Joe K, Vijesh, Jennifer S. Raj User Recommendation System Dependent on Location-Based Orientation Context. *Journal of Trends in Computer Science and Smart Technology*. 2021;3-01:4–23.
7. Grabisch M., Marichal J-L., Mesiar R., Pap E. *Aggregation Functions.* Cambridge, Cambridge University Press; 2009. (Encyclopedia of Mathematics and its Applications).
8. Yang F, Yang Zh, Cohen W. Differentiable learning of logical rules for reasoning in the knowledge base. *Advances in the field of neural information processing systems*. 2017;3:2320–2329.
9. Lyutikova L.A. Construction of a Logical-Algebraic Corrector to Increase the Adaptive Properties of the ΣΠ-Neuron. *Journal of Mathematical Sciences.* 2021;253:539–546.
10. Дюкова Е.В., Журавлев Ю.И., Прокофьев П.А. Методы повышения эффективности логических корректоров. *Машинное обучение и анализ* данных. 2015;11-1:1555–1583.

REFERENCES

1. Shibzukhov Z. Correct algorithms for aggregation of operations. *Pattern recognition and image analysis*. 2014;24-3:377–382. (In Russ.).
2. Averkin A.N., Yarushev S.A. Obzor issledovanii v oblasti razrabotki metodov izvlecheniya pravil iz iskusstvennykh neironnykh setei. *Izvestiya RAN. Teoriya i sistemy upravleniya.* 2021;6:106–121. (In Russ.).
3. Ashley I., Naimi Laura B., Balzer Multilevel Ashley I. Naimi, Laura B. Balzer. Multilevel generalization: an introduction to super learning. *European Journal of Epidemiology*. 2018;33:459–464.
4. Haoxiang W., Smith S. Big data analysis and perturbation using a data mining algorithm. *Journal of Soft Computing Paradigm*. 2021;3-01:19–28.
5. Ribeiro, M. T., Singh, S., Guestrin C. "Why should I trust you?": Explaining the predictions of any classifier. *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 2016:1135–1144.
6. Joe M., Vijesh, Jennifer S. Raj. User Recommendation System Dependent on Location-Based Orientation Context. *Journal of Trends in Computer Science and Smart Technology*. 2021;3-01:14–23.
7. Grabisch M., Marichal J-L., Mesiar R., Pap E. *Aggregation Functions.* Cambridge, Cambridge University Press; 2009. (Encyclopedia of Mathematics and its Applications).
8. Yang F., Yang Zh, Cohen W.W. Differentiable learning of logical rules for reasoning in the knowledge base. *Advances in the field of neural information processing systems.* 2017;3:2320–2329.
9. Lyutikova L.A. Construction of a Logical-Algebraic Corrector to Increase the Adaptive Properties of the ΣΠ-Neuron. *Journal of Mathematical Sciences.* 2021;253:539–546.
10. Dyukova E.V., Zhuravlev Yu.I., Prokofev P.A. Methods for increasing the efficiency of logic correctors. *Mashinnoe obuchenie i analiz dannykh*. 2015;11-1:1555–1583. (In Russ.).

ИНФОРМАЦИЯ ОБ АВТОРЕ / INFORMATION ABOUT THE AUTHOR

Лютикова Лариса Адольфовна, кандидат физико-математических наук, заведующий отделом нейроинформатики и машинного обучения. Институт прикладной математики и автоматизации КБНЦ РАН, Нальчик, Российская Федерация.
e-mail: lylarisa@yandex.ru
ORCID: 0000-0003-4941-7854

Larisa A. Lyutikova, Candidate of Physical and Mathematical Sciences, Head of the Department of Neuroinformatics and Machine Learning, Institute of Applied Mathematics and Automation KBSC RAS, Nalchik, the Russian Federation.

Статья поступила в редакцию 21.11.2023; одобрена после рецензирования 15.12.2023; принята к публикации 29.12.2023.

The article was submitted 21.11.2023; approved after reviewing 15.12.2023; accepted for publication 29.12.2023.