УДК 004.89
DOI: 10.15593/2224-9397/2021.4.07

В.В. Бахтин

Пермский национальный исследовательский политехнический университет,
Пермь, Россия

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ ДЛЯ УСТРОЙСТВ НА ПЛИС И МИКРОКОНТРОЛЛЕРАХ, ОРИЕНТИРОВАННЫХ НА ТУМАННЫЕ ВЫЧИСЛЕНИЯ

Современным проектам, использующим нейронные сети, было бы полезно разгрузить вычислительные центры, распределив вычислительные мощности для нейросетевого распознавания в распределенной сети. **Целью исследования** является разработка метода синтеза устройств реализации искусственных нейронных сетей на ПЛИС и микроконтроллерах, ориентированных на туманные вычисления. Основой для создания рассматриваемых устройств будет являться искусственная нейронная сеть, которую потребуется разделить на несколько блоков. Каждый из этих вычислительных блоков будет исполняться на отдельном физическом устройстве, связь между ними будет осуществляться с помощью стандартных каналов и протоколов. **Методика исследования** базируется на анализе информации о существующих нейронных сетях и математическом моделировании нейронной сети, которая будет пригодной для работы в режиме туманных вычислений. В **результате исследования** планируется получить математическую модель нейронной сети и метод деления нейронной сети на блоки, которые будут работать на оконечных устройствах, также будет проведено испытание данного метода на тестовом каскаде вычислительных устройств. В статье рассмотрены существующие классы искусственных нейронных сетей, исходя из полученного обзора выбраны параметры сетей, с которыми будет проводиться работа в рамках данного исследования. Разработана математическая модель, которая позволяет из обычной нейронной сети, которая монопольно выполняется на одном вычислительном устройстве, получить набор блоков, последовательное исполнение которых на каскаде устройств приведет к результатам, аналогичным результатам работы оригинальной сети. Выбраны входные параметры для метода разделения, которые позволят осуществить дальнейшие эксперименты с устройствами.

Ключевые слова: математическая модель, искусственная нейронная сеть, ПЛИС, микроконтроллеры, туманные вычисления, принятие решений, устройство нейросетевого распознавания, метод синтеза.

V.V. Bakhtin

Perm National Research Polytechnic University, Perm, Russian Federation

MATHEMATICAL MODEL OF AN ARTIFICIAL NEURAL NETWORK FOR FPGA DEVICES AND MICROCONTROLLERS FOCUSED ON FOG COMPUTING

It would be useful for modern projects using neural networks to unload computing centers by distributing computing power for neural network recognition in a distributed network. The aim of the study is to develop a method for synthesizing devices for implementing artificial neural networks on FPGAs and microcontrollers focused on fog computing. The basis for the creation of the devices in question will be an artificial neural network, which will need to be divided into several blocks. Each of these computing units will be executed on a separate physical device, communication between them will be carried out using standard channels and protocols. The research methodology is based on the analysis of information about existing neural networks and mathematical modeling of a neural network that will be suitable for operation in the fog computing mode. As a result of the research, it is planned to obtain a mathematical model of a neural network and a method for dividing a neural network into blocks that will work on terminal devices, and this method will also be tested on a test cascade of computing devices. The article considers the existing classes of artificial neural networks, based on the review obtained, the parameters of the networks with which work will be carried out within the framework of this study are selected. A mathematical model has been developed that allows a set of blocks to be obtained from a conventional neural network that is exclusively executed on one computing device, the sequential execution of which on a cascade of devices will lead to results similar to the results of the original network. Input parameters for the separation method have been selected, which will allow further experiments with devices.

Keywords: mathematical model, artificial neural network, FPGA, microcontrollers, fog computing, decision making, neural network recognition device, synthesis method.

Введение

В эпоху Интернета вещей и быстрого развития искусственного интеллекта, наблюдается быстрый рост высоко децентрализованных и интеллектуальных решений. Интернет вещей обычно производит огромное количество данных, которые для эффективного анализа, особенно в нейронных сетях, требуют высоких вычислительных возможностей. Это создает предпосылки для создания нового поколения туманных вычислений, которые поддерживают искусственный интеллект и используют архитектуру, подходящую для интеллектуальных решений. В **результате исследования** планируется получить математическую модель нейронной сети и метод деления нейронной сети на блоки, которые будут работать на оконечных устройствах, также будет проведено испытание данного метода на тестовом каскаде вычислительных устройств. Данный подход важен тем, что позволяет отойти от оптимизации сети в пользу разделения вычислительной нагрузки между устройствами.

Для решения различных задач, связанных с нейросетевым анализом данных, необходимо иметь возможность строить сложные нейросетевые структуры, обучать их на базах данных предварительно размеченных образцов и использовать обученные нейронные сети для классификации и иных задач [1]. Для обучения и использования нейронных сетей требуются большие вычислительные ресурсы, которые чаще всего имеют высокую стоимость, большое энергопотребление и должны быть физически сконцентрированы в конкретной точке пространства. У данной проблемы могут быть различные решения, например, использование облачных вычислительных ресурсов удаленных датацентров. Но использование данных ресурсов также чаще всего требует внесения денежных средств в качестве оплаты. Возможным решением данной проблемы, которое мы хотели бы предложить в данной работе, является разделение монолитной нейронной сети на каскад блоков, последовательно выполняемых на связанных между собой вычислителях. Раскроем предполагаемое решение в терминах предметной области.

Нейронная сеть – последовательность из нескольких слоев математических нейронов, соединенных друг с другом. Первый слой нейронов называется входным слоем, он анализирует данные об исследуемом объекте и переводит это в вид коэффициентов, ограниченных определенным интервалом. Последний слой нейронов называется выходным слоем, именно он передает во внешнюю среду информацию о решении, принятом нейронной сетью. Слои между входным и выходным слоем называются промежуточными или скрытыми, именно они проделывают основную работу по классификации какого-либо объекта [2]. Нейронную сеть, все слои которой производят свои вычисления на одном и том же вычислительном устройстве, назовем **монолитной нейронной сетью** (например, персептрона Розенблатта [3]). Если перед нами стоит задача получения нейронной сети, которая может проводить свои вычисления на нескольких связанных между собой устройствах, то нам потребуется разделить слои этой нейронной сети на блоки последовательных, идущих друг за другом слоев. Каждый из этих блоков будет выполнен на отдельном вычислительном устройстве, а промежуточные результаты будут переданы по сети между ними. Нейронную сеть, разбитую на набор подобных блоков, назовем **блочной нейронной сетью.** Именно метод преобразования монолитной нейронной сети к виду блочной нейронной сети, адаптированной для выполнения туманных вычислений [4], и является ключом к созданию каскадов из нескольких

не очень мощных вычислительных устройств на ПЛИС и микроконтроллерах, которые будут сопоставимы с более производительными вычислительными устройствами и иметь достаточную производительность для работы нейронных сетей. Возможно получится воспользоваться генераторами логических функций для непосредственной реализации алгоритма, например, на базовых матричных кристаллах [5, 6].

Изначально все вычисления производились на одном и том же вычислительном устройстве. Со временем скорость передачи информации начинает возрастать, становится возможным передавать по сети крупные массивы данных за короткое время, после этого появляется концепция облачных вычислений. Облачные вычисления – это вычисления, которые проводятся на удаленном компьютере, который физически не находится в прямом доступе пользователя [7]. Стало возможным с маломощной электронно-вычислительной машины, которая выступала теперь посредником при работе с мощным сервером, производить сложные и объемные вычисления. Следующим шагом развития распределенных вычислительных технологий становится концепция туманных вычислений.

Существует множество промежуточных и конечных вычислителей с небольшой производительностью, большую часть времени работы центральные процессоры этих устройств простаивают или работают с малой загрузкой, это происходит в связи с их работой в периодическом режиме. Туманные вычисления – это метод распределения вычислительных задач в виде небольших блоков на небольшие устройства, которые обрабатывают информацию в процессе ее передачи от отправителя к получателю [8]. Именно на такие блоки предлагается разбивать монолитную нейронную сеть, чтобы выполнение ее вычислений стало возможным на небольших промежуточных устройствах. Это позволит освободить ресурсы центрального вычислительного узла для других вычислительных задач [9]. Исходя из представленных предположений, можно сформулировать **цель представленной работы:** разработка метода синтеза устройств реализации искусственных нейронных сетей на ПЛИС и микроконтроллерах, ориентированных на туманные вычисления.

1. Существующие искусственные нейронные сети

Математический нейрон МакКаллока – Питса [10] представляет из себя элемент, который вычисляет выходной сигнал (по определенному правилу) из совокупности входных сигналов. Между собой нейроны мо-

гут быть соединены по-разному, но суть работы нейронной сети остается постоянной. По совокупности поступающих на вход сети сигналов на выходе формируются выходные сигналы. Нейронную сеть можно представить в виде черного ящика, у которого есть входы и выходы [11, 12].

Чаще всего структура связей между нейронами представляется в виде матрицы W, которую называют весовой матрицей. Элемент матрицы w_{ij} определяет вес связи, идущей от элемента i к элементу j. Давайте рассмотрим нейронную сеть, которая была создана в рамках предыдущего исследования при создании программного комплекса TSBuilder (рис. 1).

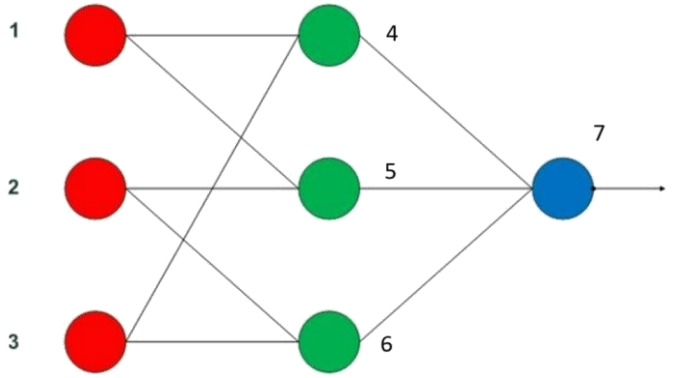

Рис. 1. Оптимизированная нейронная сеть TSBuilder

Данная сеть использовалась для классификации сложной терминологии в случае, когда термин состоял из трех слов. Нейроны входного слоя давали информацию о том, к какому полю относятся слова, входящие в термин. Нейроны скрытого слоя отвечали за категоризацию пар слов в словосочетании, а нейрон выходного слоя осуществлял голосование между попарными сравнениями и принимал решение об общей принадлежности полученного термина [13–18].Весовая матрица такой нейронной сети будет иметь следующий вид:

$$W = \begin{bmatrix} 0 & 0 & 0 & W_{14} & W_{15} & 0 & 0 \\ 0 & 0 & 0 & 0 & W_{25} & W_{26} & 0 \\ 0 & 0 & 0 & W_{34} & 0 & W_{36} & 0 \\ W_{14} & 0 & W_{34} & 0 & 0 & 0 & W_{47} \\ W_{15} & W_{25} & 0 & 0 & 0 & 0 & W_{57} \\ 0 & W_{26} & W_{36} & 0 & 0 & 0 & W_{67} \\ 0 & 0 & 0 & W_{47} & W_{57} & W_{67} & 0 \end{bmatrix}. \quad (1)$$

Например, от второго элемента к пятому идет связь, вес которой равен W_{25}. В роли функций активации могут выступать любые математические или логические функции.

Классификация нейронных сетей по характеру обучения делит их на несколько типов. Обучение с учителем предполагает, что для каждого входного вектора существует целевой вектор, представляющий собой требуемый выход. Вместе они называются обучающей парой [19]. Развитая Кохоненом [20] и многими другими модель обучения без учителя не нуждается в целевом векторе для выходов и, следовательно, не требует сравнения с предопределенными идеальными ответами. Обучающее множество состоит лишь из входных векторов. Обучающий алгоритм подстраивает веса сети так, чтобы получались согласованные выходные векторы, т. е. чтобы предъявление достаточно близких входных векторов давало одинаковые выходы.

Классификация нейронных сетей по типу настройки весов делит их на сети с фиксированными весами связей (весовые коэффициенты выбираются сразу) и сети с динамическими весами связей (обучение меняет синаптические веса) [21].

Классификация нейронных сетей по типу входной информации делит их на аналоговые (входная информация представлена в форме действительных чисел) и двоичные (входная информация в двоичном виде) [22].

Также можно классифицировать нейронные сети по тому, было ли у них предварительное обучение или они начнут свое обучение с момента начала функционирования. В данной статье описывается работа с уже обученными нейронными сетями, т.е. к моменту разделения на блочные нейронные сети веса синапсов уже будут известны и зафиксированы. Как было отмечено ранее, при построении предсказательных моделей исходные данные обычно разбиваются на обучающую и контрольную выборки. Обучающая выборка используется для обучения модели, тогда как контрольная выборка служит для получения оценки прогнозных свойств модели на новых данных [23].

В описываемом исследовании важно, чтобы к моменту синтеза устройства нейросетевого распознавания обучение было завершено и веса синапсов стабилизированы, это ограничение временное, и его можно преодолеть в будущем. Также в приведенном исследовании рассматриваются именно многослойные нейронные сети, так как в полных нейронных сетях слишком много связей, и их сложно разде-

лить на независимые кластеры, которые могли бы выполняться на различных физических устройствах. В данной статье будет рассматриваться математическая модель разделения нейронной сети без обратных связей, в будущем, в рамках исследования, возможна доработка, которая позволит разделять сети с обратными связями.

2. Математическая модель искусственной нейронной сети, ориентированной на туманные вычисления.

Дано: монолитная многослойная нейронная сеть X, результат работы которой – последовательность сигналов $\left\{y_0^K,...,y_{H_K}^K\right\}$.

Найти: последовательность нейронных сетей $\left\{\bar{X}_0,...,\bar{X}_{D-1}\right\}$, где результат вычисления $\bar{X}_1 \rightleftarrows \bar{X}_2 \rightleftarrows ... \rightleftarrows \bar{X}_{D-1}$ совпадает с результатом работы сети X.

Процесс преобразования монолитной ИНС в каскад блочных ИНС, результат вычисления которого совпадает с результатами монолитной нейронной сети, назовем **декомпозицией** искусственной нейронной сети. Начнем создание математической модели искусственной нейронной сети (рис. 2), ориентированной на туманные вычисления, с того, что воспользуемся общепринятыми определениями функционирования нейронных сетей и обозначим их за начальные точки наших изысканий:

$$y_j = \sum_{i=1}^{N} x_i w_{ij} \,, \tag{2}$$

где x_i – вход синапса, y_j – выход синапса, w_{ij} – вес синапса [24].

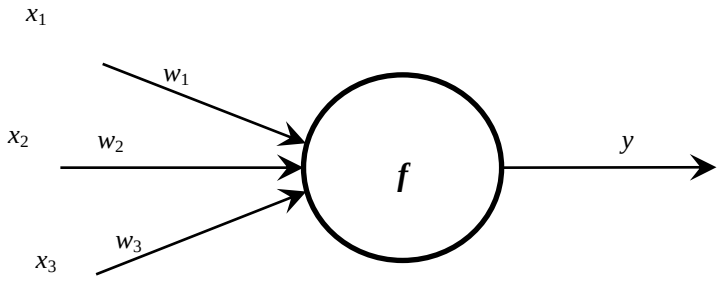

Рис. 2. Математический нейрон

Тогда функция работы ИНС на примере многослойного персептрона:

$$\begin{cases} y_i^{(k)} = f_i^{(k)}\left(\sum_{j=0}^{H_{k-1}} w_{ij}^{(k)} \cdot y_i^{(k-1)}\right), \\ y_i^{(0)} = x_i^{(0)}, \end{cases} \tag{3}$$

где $x_i^{(0)}$ – входные данные ИНС, $y_i^{(k)}$ – выход i-го нейрона k-го слоя, H_{k-1} – число нейронов на слое $k-1$, $f_i^{(k)}$ – функция активации нейрона [25].

Разделение на подсети может осуществляться с различными параметрами. Начнем с того, что рассмотрим простой пример деления монолитной нейронной сети на блочные нейронные сети с одинаковым количеством слоев в каждой из них.

Количество слоев обозначим K, количество устройств – D. Количество подсетей – D, по числу устройств. Для случая, когда $K \geq D$ и K кратно D, получим следующие рекуррентные формулы блочных нейронных сетей для каждого из устройств:

$$\bar{X}_0 : \begin{cases} y_i^{\left(\frac{K}{D}\right)} = f_i^{\left(\frac{K}{D}\right)}\left(\sum_{j=0}^{H_{\frac{K}{D}-1}} w_{ij}^{\left(\frac{K}{D}\right)} \cdot y_i^{\left(\frac{K}{D}-1\right)}\right), \\ \cdots \\ y_i^{(0)} = x_i^{(0)}, \end{cases}$$

$$\bar{X}_1 : \begin{cases} y_i^{\left(\frac{2K}{D}\right)} = f_i^{\left(\frac{2K}{D}\right)}\left(\sum_{j=0}^{H_{\frac{2K}{D}-1}} w_{ij}^{\left(\frac{2K}{D}\right)} \cdot y_i^{\left(\frac{2K}{D}-1\right)}\right), \\ \cdots \\ y_i^{\left(\frac{K}{D}+1\right)} = y_i^{\left(\frac{K}{D}\right)}, \end{cases}$$

$\cdots$

$$\bar{X}_{D-1} : \begin{cases} y_i^{(K)} = f_i^{(K)}\left(\sum_{j=0}^{H_{K-1}} w_{ij}^{(K)} \cdot y_i^{(K-1)}\right), \\ \cdots \\ y_i^{\left(\frac{(D-1)K}{D}+1\right)} = y_i^{\left(\frac{(D-1)K}{D}\right)}. \end{cases} \tag{4}$$

Обобщив данные формулы, получим общую рекуррентную формулу описания каждой из полученных блочных нейронных сетей:

$$\bar{X}_N : \begin{cases} y_i^{\left(\frac{(N+1)K}{D}\right)} = f_i^{\left(\frac{(N+1)K}{D}\right)}\left(\sum_{j=0}^{H_{\frac{(N+1)K}{D}-1}} w_{ij}^{\left(\frac{(N+1)K}{D}\right)} \cdot y_i^{\left(\frac{(N+1)K}{D}-1\right)}\right), \\ \dots \\ \begin{cases} y_i^{\left(\frac{NK}{D}+1\right)} = y_i^{\left(\frac{NK}{D}\right)}, N > 0, \\ y_i^{(0)} = x_i^{(0)}, N = 0. \end{cases} \end{cases} \tag{5}$$

Тогда, если мы обозначим послойные размерности блочных искусственных нейронных сетей как массив $\{L_0, ..., L_{D-1}\}$, то общая рекуррентная формула описания каждой из полученных блочных нейронных сетей будет следующей:

$$\begin{cases} \begin{cases} y_i^{(K)} = f_i^{(K)}\left(\sum_{j=0}^{H_{K-1}} w_{ij}^{(K)} \cdot y_i^{(K-1)}\right), N = D-1, \\ y_i^{\left(\sum_{k=0}^{N} L_k\right)} = f_i^{\left(\sum_{k=0}^{N} L_k\right)}\left(\sum_{j=0}^{H_{\sum_{k=0}^{N} L_k - 1}} w_{ij}^{\left(\sum_{k=0}^{N} L_k\right)} \cdot y_i^{\left(\sum_{k=0}^{N} L_k - 1\right)}\right), N < D-1, \end{cases} \\ \dots \\ \begin{cases} y_i^{(L_{N-1}+1)} = y_i^{(L_{N-1})}, N > 0, \\ y_i^{(0)} = x_i^{(0)}, N = 0. \end{cases} \end{cases} \tag{6}$$

Существуют различные варианты входных параметров, в зависимости от которых будут реализовываться конкретные способы разделения монолитной ИНС на части:

1) Разделение на блоки с равным количеством слоев на узле. Выше рассматривался случай, когда количество слоев нейронной сети K кратно числу устройств D, которые будут задействованы в цепочке вычисления результатов блочной нейронной сети. Для случаев, когда K не кратно D, количество слоев L_i, которые должны быть переданы на устройство D_i, вычисляется по формуле:

$$L_i = \left|\frac{K}{D}\right|. \tag{7}$$

Поскольку применяется округление в меньшую сторону для получения целого числа, количество слоев нейронной сети, которое будет на последнем вычислителе с номером $D-1$, вычисляется по формуле:

$$L_{D-1} = K - (D-1)L_0. \tag{8}$$

Таким образом, получим общую рекуррентную формулу описания каждой из полученных блочных нейронных сетей:

$$\bar{X}_N : \begin{cases} \begin{cases} y_i^{(K)} = f_i^{(K)}\left(\sum_{j=0}^{H_{K-1}} w_{ij}^{(K)} \cdot y_i^{(K-1)}\right), N = D-1, \\ y_i^{\left((N+1)\left\lfloor \frac{K}{D} \right\rfloor\right)} = f_i^{\left((N+1)\left\lfloor \frac{K}{D} \right\rfloor\right)}\left(\sum_{j=0}^{H_{(N+1)\left\lfloor \frac{K}{D} \right\rfloor - 1}} w_{ij}^{\left((N+1)\left\lfloor \frac{K}{D} \right\rfloor\right)} \cdot y_i^{\left((N+1)\left\lfloor \frac{K}{D} \right\rfloor - 1\right)}\right), N < D-1, \end{cases} \\ \cdots \\ \begin{cases} y_i^{\left(N\left\lfloor \frac{K}{D} \right\rfloor + 1\right)} = y_i^{\left(N\left\lfloor \frac{K}{D} \right\rfloor\right)}, N > 0, \\ y_i^{(0)} = x_i^{(0)}, N = 0. \end{cases} \end{cases} \tag{9}$$

Ответ: общая рекуррентная формула описания каждой из полученных блочных нейронных сетей, где массив $\{L_0, ..., L_{D-1}\}$ – количество слоев нейронной сети, которые должны быть переданы на устройство с номером N:

$$\bar{X}_N : \begin{cases} \begin{cases} y_i^{(K)} = f_i^{(K)}\left(\sum_{j=0}^{H_{K-1}} w_{ij}^{(K)} \cdot y_i^{(K-1)}\right), N = D-1, \\ y_i^{\left(\sum_{k=0}^{N} L_k\right)} = f_i^{\left(\sum_{k=0}^{N} L_k\right)}\left(\sum_{j=0}^{H_{\sum_{k=0}^{N} L_k - 1}} w_{ij}^{\left(\sum_{k=0}^{N} L_k\right)} \cdot y_i^{\left(\sum_{k=0}^{N} L_k - 1\right)}\right), N < D-1, \end{cases} \\ \cdots \\ \begin{cases} y_i^{(L_{N-1}+1)} = y_i^{(L_{N-1})}, N > 0, \\ y_i^{(0)} = x_i^{(0)}, N = 0. \end{cases} \end{cases} \tag{10}$$

2) Разделение пропорционально производительности устройств (слои). Для разделения пропорционально производительности (по количеству слоев нейронной сети) нужно сделать массив мощностей $\{P_0, ..., P_{D-1}\}$, выраженных в некоторых абсолютных единицах (результатах тестов производительности, тактовых частотах, объемах оперативной

памяти). Процент от всех слоев сети, который должен быть передан на устройство D_i для выполнения вычислений, рассчитывается по формуле:

$$\gamma_i = \frac{P_i}{\sum_{h=0}^{D-1} P_h}. \tag{11}$$

Следовательно, количество слоев L_i, которые должны быть переданы на устройство D_i, вычисляется по формуле:

$$L_i = \left\lfloor \frac{K \cdot P_i}{\sum_{h=0}^{D-1} P_h} \right\rfloor. \tag{12}$$

Поскольку применяется округление в меньшую сторону для получения целого числа, количество слоев нейронной сети, которое будет на последнем вычислителе с номером $D-1$, вычисляется по формуле:

$$L_{D-1} = K - \sum_{h=0}^{D-2} L_h. \tag{13}$$

Обобщив формулы (12), (13) и (10), получим общую рекуррентную формулу описания каждой из полученных блочных нейронных сетей:

$$\overline{X}_N : \begin{cases} \begin{cases} y_i^{(K)} = f_i^{(K)}\left(\sum_{j=0}^{H_{K-1}} w_{ij}^{(K)} \cdot y_i^{(K-1)}\right), N = D-1, \\ y_i^{\left(\sum_{m=0}^{N}\left\lfloor \frac{K\cdot P_m}{\sum_{h=0}^{D-1} P_h}\right\rfloor\right)} = f_i^{\left(\sum_{m=0}^{N}\left\lfloor \frac{K\cdot P_m}{\sum_{h=0}^{D-1} P_h}\right\rfloor\right)} \times \\ \times \left(\sum_{j=0}^{H^{\sum_{m=0}^{N}\left\lfloor \frac{K\cdot P_m}{\sum_{h=0}^{D-1} P_h}\right\rfloor - 1}} w_{ij}^{\left(\sum_{m=0}^{N}\left\lfloor \frac{K\cdot P_m}{\sum_{h=0}^{D-1} P_h}\right\rfloor\right)} \cdot y_i^{\left(\sum_{m=0}^{N}\left\lfloor \frac{K\cdot P_m}{\sum_{h=0}^{D-1} P_h}\right\rfloor - 1\right)} \right), N < D-1, \end{cases} \\ \cdots \\ \begin{cases} y_i^{\left(\sum_{m=0}^{N-1}\left\lfloor \frac{K\cdot P_m}{\sum_{h=0}^{D-1} P_h}\right\rfloor + 1\right)} = y_i^{\left(\sum_{m=0}^{N-1}\left\lfloor \frac{K\cdot P_m}{\sum_{h=0}^{D-1} P_h}\right\rfloor\right)}, N > 0, \\ y_i^{(0)} = x_i^{(0)}, N = 0. \end{cases} \end{cases} \tag{14}$$

3) Разделение на блоки с равным количеством нейронов на узле.

Берем сумму всех нейронов на всех слоях монолитной нейронной сети. На каждом слое всего H_b нейронов, тогда желаемое число нейронов в каждой из блочных ИНС будет следующим:

$$\text{Neur}_N = \frac{\sum_{b=1}^{K} H_b}{D}. \quad (15)$$

В целочисленном варианте:

$$\text{Neur}_N = \left\lfloor \frac{\sum_{b=1}^{K} H_b}{D} \right\rfloor. \quad (16)$$

И в последней нейронной сети:

$$\text{Neur}_{D-1} = \sum_{b=1}^{K} H_b - (D-1)\left\lfloor \frac{\sum_{b=1}^{K} H_b}{D} \right\rfloor. \quad (17)$$

Но данный вариант тоже не подходит: нейроны разделены на слои, а делить слой, разнося его на различные устройства, нецелесообразно, так как это увеличит объем данных, пересылаемых между физическими устройствами. По этой причине разделять сеть будем послойно, а определять число нейронов и слоев на каждой из блочных ИНС – с помощью сравнения с желаемым значением Neur_N. Чтобы получить массив $\{L_0,...,L_{D-1}\}$, проведем предварительный шаг оценки, используя жадный алгоритм:

1. На очередной ИНС нет слоев нейронов.
2. Если количество нейронов на слоях суммарно со слоем-претендентом меньше, чем $\left\lfloor \frac{\sum_{b=1}^{K} H_b}{D} \right\rfloor$, то добавить слой к ИНС, т.е. увеличить L_N на единицу.
3. Иначе – зафиксировать ИНС как завершенную, перейти к составлению следующей ИНС.
4. На ИНС с номером D – 1 перенести все оставшиеся слои.

Тогда после получения массива $\{L_0,...,L_{D-1}\}$ достаточно подставить их в (10).

4) Разделение пропорционально производительности устройств (нейроны).

В способе с распределением нейронов по блочным ИНС пропорционально мощности устройств желаемое число нейронов Neur_N для ИНС будет равно:

$$\text{Neur}_N = \frac{\sum_{b=1}^{K} H_b \cdot P_N}{\sum_{h=0}^{D-1} P_h}, N \neq D-1. \quad (18)$$

В целочисленном варианте:

$$\text{Neur}_N = \left\lfloor \frac{\sum_{b=1}^{K} H_b \cdot P_N}{\sum_{h=0}^{D-1} P_h} \right\rfloor, N \neq D-1. \quad (19)$$

Тогда последняя ИНС:

$$\text{Neur}_N = \sum_{b=1}^{K} H_b \cdot (D-1) \left\lfloor \frac{\sum_{b=1}^{K} H_b \cdot P_N}{\sum_{h=0}^{D-1} P_h} \right\rfloor, N = D-1. \quad (20)$$

По той же причине, что и в предыдущем способе, делить нейронную сеть будем послойно. Определять число нейронов и слоев в сети будем с помощью сравнения с L_N по следующему алгоритму:

1. На очередной ИНС нет слоев нейронов.

2. Если количество нейронов на слоях суммарно со слоем-претендентом меньше, чем $\left\lfloor \frac{\sum_{b=1}^{K} H_b \cdot P_N}{\sum_{h=0}^{D-1} P_h} \right\rfloor$, то добавить слой к ИНС, т.е. увеличить L_N на единицу.

3. Иначе – зафиксировать ИНС как завершенную, перейти к составлению следующей ИНС.

4. На ИНС с номером D – 1 перенести все оставшиеся слои.

Тогда после получения массива $\{L_0, ..., L_{D-1}\}$ достаточно подставить их в (10).

5) Разделение с условием минимизации передаваемых по сети данных.

Для того чтобы минимизировать объем данных, передаваемых по сети между устройствами, которые будут выполнять вычисления блочных ИНС, необходимо разделять монолитную нейронную сеть на части, рассекая ее в тех местах, где число нейронов на соответствующем слое минимально. В этом случае будет минимальное число параметров, которые потребуется передавать по сети между вычислительными устройствами.

Для того чтобы получить массив послойных размерностей блочных ИНС $\{L_0,...,L_{D-1}\}$, потребуется осуществить предварительную подготовку. В первую очередь нам потребуется отобрать $D-1$ слоев нейронной сети с наименьшим количеством нейронов. Отсортируем для этого массив слоев нейронной сети по количеству нейронов на каждом слое, сортировка по возрастанию. Первые $D-1$ слоев и есть искомые нами слои, отделим их в новый массив и отсортируем уже по порядковому номеру слоя в монолитной нейронной сети, по возрастанию. Увеличим размерность массива Numb до $D+1$, на нулевое место поставим 0, на последнее – номер последнего слоя K. Полученный массив $\{\mathrm{Numb}_0,...,\mathrm{Numb}_D\}$ будет использован нами для вычисления массива послойных размерностей блочных ИНС $\{L_0,...,L_{D-1}\}$ по следующей формуле:

$$L_N = \mathrm{Numb}_{N+1} - \mathrm{Numb}_N. \qquad (21)$$

Далее от нас требуется подставить полученные значения массива $\{L_0,...,L_{D-1}\}$ в (10).

3. Предлагаемые дальнейшие шаги исследования

Первая **задача исследования** уже решена, математическая модель, подходящая для создания метода синтеза устройств нейросетевого распознавания на ПЛИС и микроконтроллерах, представлена в предыдущем разделе. Из полученных методов для непосредственно моделирования и реализации на физических устройствах была выбрана разновидность математической модели, которая описывает разделение на блоки с равным количеством слоев на узле. Выбрано разделение на блоки с такими входными ограничениями по той причине, что вычислители, которые будут использованы в дальнейших экспериментах, будут иметь одинаковые параметры вычислительной мощности.

Существуют архитектуры ИНС и варианты использования блоков, которые совпадут в каскадах различных монолитных ИНС после декомпозиции, в которых последовательность блочных ИНС будет превращаться в граф вычислений, такие примеры будут рассмотрены в следующих частях исследования.

Следующим шагом станут формализация метода синтеза устройств и реализация необходимого программного обеспечения. Уже

начата разработка программного комплекса Separator, который сможет получать из монолитной нейронной сети несколько нейронных сетей, которые будут работать на различных вычислительных устройствах, которые будут подключены последовательно, моделируя реальную вычислительную сеть.

Для моделирования работы каскада блочных нейронных сетей в рамках классических вычислительных устройств будут использованы одноплатные микрокомпьютеры Raspberry Pi 4 Model B, которые по вычислительной мощности сопоставимы с современными коммутаторами или роутерами. Соединение между микрокомпьютерами будет осуществляться с помощью проводного соединения стандарта Ethernet, хотя возможны эксперименты с беспроводными каналами Wi-Fi и Bluetooth. А для моделирования вычислений с помощью ПЛИС, которые реализованы в виде систем на кристалле, будут использованы Atmel ATMega 32, которые будут соединены между собой по проводному соединению UART или USART. Данные вычислители помогут смоделировать менее мощные датчики и микроконтроллеры, которые также могут входить в реальные вычислительные сети.

Заключение

Целью исследования является разработка метода синтеза устройств реализации искусственных нейронных сетей на ПЛИС и микроконтроллерах, ориентированных на туманные вычисления. В **результате исследования** создана математическая модель распознавания объектов с помощью искусственных нейронных сетей устройствами на ПЛИС и микроконтроллерах, ориентированными на туманные вычисления. Сформулированы математические определения монолитных и блочных искусственных нейронных сетей, описаны в математическом виде действия, которые требуется осуществить для того, чтобы из одной монолитной нейронной сети получить каскад блочных нейронных сетей с заданными параметрами. Описано несколько примеров подобных переходов с различными входными параметрами, сформулирована общая рекуррентная формула (10) для результирующих каскадов блочных нейронных сетей.

Приведены базовые понятия теории искусственных нейронных сетей, рассмотрены имеющиеся классификации нейронных сетей, выведены граничные условия проводимого исследования, т.е. проведен

выбор классов нейронных сетей, для которых будут справедливыми те теоретические и математические положения, которые представлены в данной статье. Определены параметры нейронных сетей, с которыми будут произведены эксперименты по синтезу устройств нейросетевого распознавания. Создание устройств нейросетевого распознавания сможет значительно сократить нагрузку на серверы распознавания (ориентировочно на десятки процентов) и увеличить загруженность устройств в сети, задействовав их простаивающие мощности (с 5–15 до 60–70 %).

Исследование проводится при поддержке РФФИ на средства гранта № 20-37-90036 – Аспиранты «Метод синтеза устройств нейросетевого распознавания для реализации режима Fog computing».

Библиографический список

1. Бахтин В.В. Модификация алгоритма идентификации и категоризации научных терминов с использованием нейронной сети // Нейрокомпьютеры: разработка, применение. – 2019. – Т. 21, № 3. – С. 14–19.

2. Zupan Jure. Introduction to Artificial Neural Network (ANN) Methods: What They Are and How to Use Them // Acta Chimica Slovenica. – 1994. – Vol. 41, № 3. – P. 327–352.

3. Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain // Psychological Review. – 1958. – Vol. 65, № 6. – P. 386–408. DOI: 10.1037/h0042519

4. Aazam, Mohammad, Zeadally, Sherali, Harras, Khaled Fog Computing Architecture, Evaluation, and Future Research Directions // IEEE Communications Magazine. – 2018. – № 56. – P. 46–52. DOI: 10.1109/MCOM.2018.1700707

5. Тюрин С.Ф., Каменских А.Н. БМК-реализация самосинхронного генератора логических функций для нейронных сетей // Нейрокомпьютеры: разработка, применение. – 2018. – № 7. – С. 26–32.

6. Тюрин С.Ф. Анализ дискретных пороговых элементов нейронный сетей // Нейрокомпьютеры: разработка, применение. – 2018. – № 7. – С. 20–25.

7. Surbiryala J., Rong C. Cloud Computing: History and Overview // IEEE Cloud Summit. – 2019. – P. 1–7. DOI: 10.1109/CloudSummit47114.2019.00007

8. Fog Computing: A Comprehensive Architectural Survey / P. Habibi, M. Farhoudi, S. Kazemian, S. Khorsandi, A. Leon-Garcia // IEEE Access. – 2020. – Vol. 8. – P. 69105–69133. DOI: 10.1109/ACCESS.2020.2983253

9. Priyabhashana H.M.B., Jayasena K.P.N. Data Analytics with Deep Neural Networks in Fog Computing Using TensorFlow and Google Cloud Platform // 14th Conference on Industrial and Information Systems (ICIIS). – 2019. – P. 34–39. DOI: 10.1109/ICIIS47346.2019.9063284

10. Hayman S. The McCulloch-Pitts model // International Joint Conference on Neural Networks. Proceedings (Cat. No.99CH36339). – 1999. – Vol. 6. – P. 4438–4439. DOI: 10.1109/IJCNN.1999.830886

11. Гафаров Ф.М., Галимянов А.Ф. Искусственные нейронные сети и приложения: учеб. пособие. – Казань: Изд-во Казан. ун-та, 2018. – С. 121.

12. Галимянов Ф.А., Гафаров Ф.М., Хуснутдинов Н.Р. Модель роста нейронной сети // Математическое моделирование. – 2011. – Т. 23, № 3. – С. 101–108.

13. Isaeva E., Bakhtin V., Tararkov A. Collecting the Database for the Neural Network Deep Learning Implementation / Antipova T., Rocha A. (eds.) Digital Science // DSIC18 2018. Advances in Intelligent Systems and Computing. – Springer, Cham, 2019. – Vol. 850. – P. 12–18. DOI: 10.1007/978-3-030-02351-5_2

14. Bakhtin V.V., Isaeva E.V. New TSBuilder: Shifting towards Cognition // 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus). – 2019. – P. 179–181. DOI: 10.1109/EIConRus.2019.8656917

15. Bakhtin V.V., Isaeva E.V., Tararkov A.V. TSBuilder 2.0: Improving the Identification Accuracy Due to Synonymy // 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus). – 2020. – P. 225–228. DOI: 10.1109/EIConRus49466.2020.9039207

16. Bakhtin V.V., Isaeva E.V., Tararkov A.V. TSMiner: from TSBuilder to Ecosystem // 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus). – 2021. – P. 221–224. DOI: 10.1109/ElConRus51938.2021.9396569

17. Isaeva E., Bakhtin V., Tararkov A. Formal Cross-Domain Ontologization of Human Knowledge / Rocha Á., Ferrás C., Montenegro

Marin C., Medina García V. (eds) // Information Technology and Systems. ICITS 2020. Advances in Intelligent Systems and Computing. – Springer, Cham, 2020. – Vol. 1137. – P. 94–103. DOI: 10.1007/978-3-030-40690-5_10

18. Bakhtin V., Isaeva E. Developing an Algorithm for Identification and Categorization of Scientific Terms in Natural Language Text through the Elements of Artificial Intelligence // 14th International Scientific-Technical Conference on Actual Problems of Electronic Instrument Engineering (APEIE) – 44894. Proceedings. – Novosibirsk, 2018. – P. 384–390.

19. Tang Z., Wang D., Zhang Z. Recurrent neural network training with dark knowledge transfer // IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). – 2016. – P. 5900–5904. DOI: 10.1109/ICASSP.2016.7472809

20. Kohonen T. The self-organizing map // Proceedings of the IEEE. – 1990. – Vol. 78, № 9. – P. 1464–1480. DOI: 10.1109/5.58325

21. Hubert A.B., Te Braake, Gerrit Van Straten Random activation weight neural net (RAWN) for fast non-iterative training // Engineering Applications of Artificial Intelligence. – 1995. – Vol. 8, iss. 1. – P. 71–80. DOI: 10.1016/0952-1976(94)00056-S

22. Каменских А.Н., Тюрин С.Ф. Методика комбинированного резервирования асинхронных нейронных сетей // Нейрокомпьютеры: разработка, применение. – 2016. – № 8. – С. 36–40.

23. Fast Deep Neural Networks With Knowledge Guided Training and Predicted Regions of Interests for Real-Time Video Object Detection / W. Cao, J. Yuan, Z. He, Z. Zhang, Z. He // IEEE Access. – 2018. – Vol. 6. – P. 8990–8999. DOI: 10.1109/ACCESS.2018.2795798

24. Yasnitsky L.N., Yasnitsky V.L. Technique of design of integrated economic and mathematical model of mass appraisal of real estate property by the example of Yekaterinburg housing market // Journal of Applied Economic Sciences. – 2016. – Vol. 11, no. 8. – P. 1519–1530.

25. Ясницкий Л.Н. Интеллектуальные системы. – М.: Лаборатория знаний, 2016.

References

1. Bakhtin V.V. Modifikatsiia algoritma identifikatsii i kategorizatsii nauchnykh terminov s ispol'zovaniem neironnoi seti [Modification of the algorithm for identification and categorization of scientific terms using a

neural network]. *Neirokomp'iutery: razrabotka, primenenie*, 2019, vol. 21, no. 3, pp. 14-19.

2. Zupan Jure. Introduction to Artificial Neural Network (ANN) Methods: What They Are and How to Use Them. *Acta Chimica Slovenica*, 1994, vol. 41, no. 3, pp. 327-352.

3. Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 1958, vol. 65, no. 6, pp. 386-408. DOI: 10.1037/h0042519

4. Aazam, Mohammad, Zeadally, Sherali, Harras, Khaled Fog Computing Architecture, Evaluation, and Future Research Directions. *IEEE Communications Magazine*, 2018, no. 56, pp. 46-52. DOI: 10.1109/MCOM.2018.1700707

5. Tiurin S.F., Kamenskikh A.N. BMK-realizatsiia samosinkhronnogo generatora logicheskikh funktsii dlia neironnykh setei [BMK-implementation of a self-timed logical function generator for neural networks]. *Neirokomp'iutery: razrabotka, primenenie*, 2018, no. 7, pp. 26-32.

6. Tiurin S.F. Analiz diskretnykh porogovykh elementov neironnyi setei [Analysis of discrete threshold elements of neural networks]. *Neirokomp'iutery: razrabotka, primenenie*, 2018, no. 7, pp. 20-25.

7. Surbiryala J., Rong C. Cloud Computing: History and Overview. *IEEE Cloud Summit*, 2019, pp. 1-7. DOI: 10.1109/CloudSummit47114.2019.00007

8. Habibi P., Farhoudi M., Kazemian S., Khorsandi S., Leon-Garcia A. Fog Computing: A Comprehensive Architectural Survey. *IEEE Access*, 2020, vol. 8, pp. 69105-69133. DOI: 10.1109/ACCESS.2020.2983253

9. Priyabhashana H.M.B., Jayasena K.P.N. Data Analytics with Deep Neural Networks in Fog Computing Using TensorFlow and Google Cloud Platform. *14th Conference on Industrial and Information Systems (ICIIS)*, 2019, pp. 34-39. DOI: 10.1109/ICIIS47346.2019.9063284

10. Hayman S. The McCulloch-Pitts model. *International Joint Conference on Neural Networks. Proceedings (Cat. No.99CH36339)*, 1999, vol. 6, pp. 4438-4439. DOI: 10.1109/IJCNN.1999.830886

11. Gafarov F.M., Galimianov A.F. Iskusstvennye neironnye seti i prilozheniia [Artificial neural networks and applications]. Kazan': Kazanskii universitet, 2018, 121 p.

12. Galimianov F.A., Gafarov F.M., Khusnutdinov N.R. Model' rosta neironnoi seti [The growth model of the neural network]. *Matematicheskoe modelirovanie*, 2011, vol. 23, no. 3, pp. 101-108.

13. Isaeva E., Bakhtin V., Tararkov A. Collecting the Database for the Neural Network Deep Learning Implementation. Antipova T., Rocha A. (eds.) Digital Science. *DSIC18 2018. Advances in Intelligent Systems and Computing*. Springer, Cham, 2019, vol. 850, pp. 12-18. DOI: 10.1007/978-3-030-02351-5_2

14. Bakhtin V.V., Isaeva E.V. New TSBuilder: Shifting towards Cognition. *2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus*, 2019, pp. 179-181. DOI: 10.1109/EIConRus.2019.8656917

15. Bakhtin V.V., Isaeva E.V., Tararkov A.V. TSBuilder 2.0: Improving the Identification Accuracy Due to Synonymy. *2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, 2020, pp. 225-228. DOI: 10.1109/EIConRus49466.2020.9039207

16. Bakhtin V.V., Isaeva E.V., Tararkov A.V. TSMiner: from TSBuilder to Ecosystem. *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, 2021, pp. 221-224. DOI: 10.1109/ElConRus51938.2021.9396569

17. Isaeva E., Bakhtin V., Tararkov A. Formal Cross-Domain Ontologization of Human Knowledge. Rocha Á., Ferrás C., Montenegro Marin C., Medina García V. (eds.). *Information Technology and Systems. ICITS 2020. Advances in Intelligent Systems and Computing*. Springer, Cham, 2020, vol. 1137, pp. 94-103. DOI: 10.1007/978-3-030-40690-5_10

18. Bakhtin V., Isaeva E. Developing an Algorithm for Identification and Categorization of Scientific Terms in Natural Language Text through the Elements of Artificial Intelligence. *14th International Scientific-Technical Conference on Actual Problems of Electronic Instrument Engineering (APEIE) - 44894. Proceedings*. Novosibirsk, 2018, pp. 384-390.

19. Tang Z., Wang D., Zhang Z. Recurrent neural network training with dark knowledge transfer. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5900-5904. DOI: 10.1109/ICASSP.2016.7472809

20. Kohonen T. The self-organizing map. *Proceedings of the IEEE*, 1990, vol. 78, no. 9, pp. 1464-1480. DOI: 10.1109/5.58325

21. Hubert A.B., Te Braake, Gerrit Van Straten Random activation weight neural net (RAWN) for fast non-iterative training. *Engineering Applications of Artificial Intelligence*, 1995, vol. 8, iss. 1, pp. 71-80. DOI: 10.1016/0952-1976(94)00056-S

22. Kamenskikh A.N., Tiurin S.F. Metodika kombinirovannogo rezervirovaniia asinkhronnykh neironnykh setei [A technique for combined redundancy of asynchronous neural networks]. *Neirokomp'iutery: razrabotka, primenenie*, 2016, no. 8, pp. 36-40.

23. Cao W., Yuan J., He Z., Zhang Z., He Z. Fast Deep Neural Networks With Knowledge Guided Training and Predicted Regions of Interests for Real-Time Video Object Detection. *IEEE Access*, 2018, vol. 6, pp. 8990-8999. DOI: 10.1109/ACCESS.2018.2795798

24. Yasnitsky L.N., Yasnitsky V.L. Technique of design of integrated economic and mathematical model of mass appraisal of real estate property by the example of Yekaterinburg housing market. *Journal of Applied Economic Sciences*, 2016, vol. 11, no. 8, pp. 1519-1530.

25. Iasnitskii L.N. Intellektual'nye sistemy [Intelligent systems]. Moscow: Laboratoriia znanii, 2016.

Сведения об авторе

Бахтин Вадим Вячеславович (Пермь, Россия) – аспирант, младший научный сотрудник кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета (614990, Пермь, Комсомольский пр., 29, e-mail: bakhtin_94@bk.ru).

About the author

Vadim V. Bakhtin (Perm, Russian Federation) – Graduate Student, Junior Researcher of the Department of Automation and Telemechanics Perm National Research Polytechnic University (614990, Perm, 29, Komsomolsky pr., e-mail: bakhtin_94@bk.ru).

Получено 15.09.2021