

Реализация элементов памяти и комбинационной логики в классе пирамидальных нейронных сетей быстрого обучения

А. Ю. Дорогов

Санкт-Петербургский государственный электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)
vaks2006@yandex.ru

Аннотация. В работе рассматривается применение пирамидальных нейронных сетей быстрого обучения для реализации элементов памяти и комбинационной логики на примере шифратора и дешифратора бинарных кодов. Сети рассматриваемого класса представимы линейными операторами, имеют самоподобную структуру и являются частным случаем алгоритма быстрого преобразования Фурье. Приведены методы топологического построения пирамидальных сетей. Представлен алгоритм быстрого обучения сетей, обладающий абсолютной сходимостью. Показано что пирамидальная сеть памяти обеспечивает хранение и точное восстановление образов подобно хранению чисел в компьютерной памяти произвольного доступа. Отмечено что пирамидальные сети относятся к категории сетей глубокого обучения.

Ключевые слова: быстрая нейронная сеть; фрактальные произведения; самоподобные структуры; образная память; логические элементы

I. ВВЕДЕНИЕ

Простые логические элементы комбинационной логики «И», «ИЛИ», «НЕТ» легко реализуются на одиночных нейронах с пороговой активационной функцией. Немного сложнее реализуется элемент «Исключающее ИЛИ», который требует несколько нейронов в двухслойной сети. Для реализации сложных логических функций возможен путь, повторяющий схемотехнику логических устройств, но это решение исключает потенциал нейронных сетей, связанный с обучением. Альтернативным вариантом является реализация сложной логики через обучение нейронной сети. Однако в этом направлении возникают проблемы целесообразного выбора структуры нейронной сети, так чтобы она была достаточной и не избыточной для реализации логических функций. Кроме того, классические методы обучения нейронных сетей, построенные на основе алгоритма с обратным распространением ошибки, не гарантируют сходимость, что добавляет неопределённость в выборе структуры сети.

Логические функции может быть реализованы также через элементы памяти. К нейронным сетям реализующим память относится большой класс нейронных сетей с обратными связями, такие как сети Элмана [1], Хопфилда [2], Хэмминга [3], АРТ [4] и другие. Эти сети решают задачи восстановления искажённых образов, ассоциативной памяти,

кратковременной динамической памяти и другие подобные. Нейронные сети данного типа, как правило, используются в контексте распознавания образов, задача точного восстановления образа при этом не ставится. Это обстоятельство усложняет реализацию логических функций, где требуется точное воспроизведение логики.

В данной работе будут рассмотрены пирамидальные с сети с регулярной самоподобной структурой. Будет показано, что сети данного типа в равной степени могут быть использованы как для реализации сложной комбинационной логики, так и для хранения и точного восстановления образов. Сети данного класса имеют глубокую связь с алгоритмом быстрого преобразования Фурье и обобщённого быстрого преобразования, для обозначения нового класса сетей используется термин быстрые нейронные сети (БНС) [5].

Для БНС можно предложить быстрые алгоритмы обучения, абсолютно сходящиеся за конечное число шагов. В основе алгоритмов обучения БНС лежит доказанное свойство структурной фрактальности, которое можно выразить системным инвариантом морфологического уровня [6]. Идея метода обучения БНС к одной или нескольким функциям основана на представлении каждой функции заданного набора в виде предфрактального произведения, отвечающего мультипликативной форме представления элементов матрицы быстрого преобразования.

II. БЫСТРЫЕ ПИРАМИДАЛЬНЫЕ НЕЙРОННЫЕ СЕТИ

В работе [7] показано, что модель алгоритма быстрого преобразования в топологии «Кули-Тьюки с прореживанием по частоте» определяется кортежами:

$$\begin{aligned} U^m &= \langle v_{m-1}v_{m-2} \dots v_0 u_{n-1}u_{n-2} \dots u_m \rangle, \\ V^m &= \langle v_m v_{m-1} \dots v_0 u_{n-1}u_{n-2} \dots u_{m+1} \rangle, \\ z^m &= \langle v_{m-1} \dots v_0 u_{n-1}u_{n-2} \dots u_{m+1} \rangle. \end{aligned} \quad (2)$$

Здесь m – номер слоя, а локальные переменные u_m, v_m определяют позиционные номера рецепторов и аксонов базовых операций в слое m . Кортежи используются для поразрядного представления чисел в позиционной системе счисления. Например, для системы счисления с основанием 2 имеем:

$$u = \langle u_{n-1}u_{n-2} \dots u_0 u_1 \rangle = u_{n-1}2^{n-1} + u_{n-2}2^{n-2} + \dots + u_1 2 + u_0.$$

Из выражения (2) следует, что для нулевого слоя номер базовой операции (далее – нейронного ядра) определяется выражением:

$$z^0 = \langle u_{n-1}u_{n-2} \dots u_1 \rangle,$$

а глобальный номер рецептора нулевого нейронного слоя выражением:

$$U^0 = \langle u_{n-1}u_{n-2} \dots u_0 \rangle. \quad (3)$$

Локальные переменные u_0, v_0 определяют позиционный номер рецептора и позиционный номер аксона в пределах каждого ядра нулевого слоя (нумерация рецепторов и аксонов начинается с нуля). Будем полагать, что значение p_0 определяет число рецепторов ядра нулевого слоя, а значение g_0 – число аксонов. Аналогичные обозначения p_m, g_m , где $m=0,1,\dots,n-1$, будем использовать для обозначения размерностей рецепторных и аксоновых полей остальных слоёв сети.

Выберем значения $g_m = 1$ для $m=1,2,\dots,n-1$, а для $m=0$ сохраним произвольное значение g_0 . В этом случае из (2) следует, что аксоны последнего слоя получат глобальные позиционные номера

$$V^{n-1} = \langle 0_{n-1}0_{n-2} \dots 0_1 v_0 \rangle.$$

Таким образом, нейронная сеть будет иметь ровно g_0 выходов, а из (3) следует, что число входов будет равно $N = p_0 \cdot p_1 \dots p_{n-1}$. Построенная сеть является пирамидальной, с размерностью N по входу и g_0 по выходу. На рис. 1 приведён пример трёхслойной пирамидальной сети для структуры $(p_0, p_1, p_2) = (4, 2, 2)$, $(g_0, g_1, g_2) = (3, 1, 1)$.

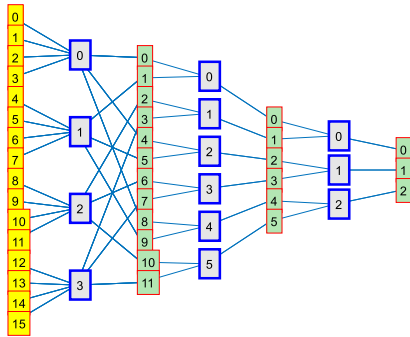


Рис. 1. Трёхслойная пирамидальная БНС с регулярной топологией

Детали построения топологического графа БНС изложены в работе [7].

Для БНС доказана обобщённая теорема факторизации [7], согласно которой элементы матрицы преобразования представимы через произведение элементов базовых операций:

$$h(U, V) = w_{z^0}^0(u_0, v_0) w_{z^1}^1(u_1, v_1) \dots w_{z^{n-2}}^{n-2}(u_{n-2}, v_{n-2}) w_{z^{n-1}}^{n-1}(u_{n-1}, v_{n-1}). \quad (4)$$

Здесь элементы матриц базовых операций $W_{z^m}^m$ обозначены через $w_{z^m}^m(u_m, v_m)$, где u_m, v_m – определяют номер строки и номер столбца матрицы, m – номер слоя, и z^m – номер базовой операции в пределах слоя.

В работе [7] приведён алгоритм мультипликативной декомпозиции произвольной функции заданной на дискретном интервале длиной $N = p_0 p_1 \dots p_{n-1}$. Мультипликативная декомпозиция имеет вид:

$$f(u) = f_0 \langle u_{n-1}u_{n-2} \dots u_0 \rangle = \phi_{i^0}^0(u_0) \phi_{i^1}^1(u_1) \dots \phi_{i^{n-2}}^{n-2}(u_{n-2}) \phi_{i^{n-1}}^{n-1}(u_{n-1}), \quad (5)$$

где $i^m = \langle u_{n-1}u_{n-2} \dots u_{m+1} \rangle$. Аргумент представляется в виде числа в многоосновной системе счисления:

$$u = \langle u_{n-1}u_{n-2} \dots u_0 \rangle = u_{n-1}p_{n-2}p_{n-3} \dots p_0 + \dots + u_{n-2}p_{n-3}p_{n-4} \dots p_0 + \dots + u_1 p_0 + u_0.$$

Здесь $u_i \in [0, 1, \dots, p_i - 1]$ – разрядные переменные. Нетрудно заметить, что выражения (4) и (5) подобны, учитывая, что для построенной пирамидальной сети

$$z^m = \langle 0_{m-1} \dots 0_1 v_0 u_{n-1} u_{n-2} \dots u_{m+1} \rangle,$$

можно записать следующее правило обучения нейронных ядер к эталонным функциям:

$$\begin{aligned} w_{z^0}^0(u_0, v_0) &= \phi_{i^0}^k(u_0), \quad m=0, \\ w_{z^m}^m(u_m, 0) &= \phi_{i^m}^k(u_m), \quad m=1, 2, \dots, n-1, \\ z^m &= \langle 0_{m-1} \dots 0_1 v_0 u_{n-1} u_{n-2} \dots u_{m+1} \rangle, \\ i^m &= \langle u_{n-1} u_{n-2} \dots u_{m+1} \rangle, \end{aligned}$$

где k – номер эталонной функции. Взаимно-однозначное соответствие $k \leftrightarrow v_0$ определяет упорядочение эталонных функций по их привязке к нейронам выходного слоя.

Архитектура быстрой пирамидальной нейронной сети легко обобщается на многомерный случай [8]. Выражения (2) для топологии сети в этом случае примут вид:

$$\begin{aligned} U^m &= \langle 0_{m-1}^* 0_{m-2}^* \dots 0_1^* v_0^* u_{n-1}^* u_{n-2}^* \dots u_m^* \rangle, \\ V^m &= \langle 0_m^* 0_{m-1}^* \dots 0_1^* v_0^* u_{n-1}^* u_{n-2}^* \dots u_{m+1}^* \rangle, \\ z^m &= \langle 0_{m-1}^* 0_{m-2}^* \dots 0_1^* v_0^* u_{n-1}^* u_{n-2}^* \dots u_{m+1}^* \rangle, \end{aligned}$$

Отсюда следует правило обучения двумерной нейронной сети:

$$\begin{aligned} w_{z^m, i_z}^m(u_m^y, u_m^x; v_m^y, v_m^x) &= f_{i_z, i_y}^k(u_m^y, u_m^x), \quad m=0, \\ w_{z^m, z_y}^m(u_m^y, u_m^x; 0_m^y, 0_m^x) &= f_{i_z, i_y}^k(u_m^y, u_m^x), \quad m>0, \\ z^m &= \langle 0_{m-1}^* 0_{m-2}^* \dots 0_1^* v_0^* u_{n-1}^* u_{n-2}^* \dots u_{m+1}^* \rangle, \\ i_z^m &= \langle u_{n-1}^* u_{n-2}^* \dots u_{m+1}^* \rangle. \end{aligned}$$

Здесь $w_{z_x, z_y}^m(u_m^y, u_m^x; v_m^y, v_m^x)$ – двумерные нейронные ядра, $f_{i_x, i_y}^k(u_m^y, u_m^x)$ – множители мультипликативного разложения эталонного изображения, x, y – пространственные координаты. Символ «*» заменяет обозначения пространственных координат. Взаимнооднозначное соответствие $k \leftrightarrow (y_m^y, v_m^x)$ определяет упорядочение эталонных функций в выходной плоскости.

III. ЛОГИЧЕСКИЙ ДЕШИФРАТОР

Пирамидальные сети быстрого обучения могут быть точно настроены на произвольные эталонные функции. Количество эталонов определяется выбранной топологией сети. Будем полагать, что настройка сети выполнена к эталонным функциям, нормированным по энергии к единице и представляющим собой логические коды по основанию 2. В рабочем режиме сеть выполняет вычисление скалярных произведений входной вектор-функции с эталонными функциями. Если входная функция нормирована по энергии к единице и совпадает с эталоном, то на соответствующем выходе нейронной сети получим единичное значение, в то время как остальные выходы будут иметь значения меньше единицы. Это обстоятельство можно использовать для построения дешифраторов логических кодов.

Рассмотрим, например, построение дешифратора для четырёхбуквенных кодовых слов. Для кода с основанием 2 дешифратор должен иметь 4 входа и 16 выходов. Нейронная сеть должна иметь такие же размерности по входу и выходу. Выберем одномерную двухслойную нейронную сеть, топология сети показана на рис. 2.

Сеть обучается к кодам, представленным в таблице на этом же рисунке. Коды рассматриваются как вектор-функции. Обучающие эталонные функции формируются из кодов нормированием по энергии к единице. Сеть имеет структурные характеристики: по рецепторным полям $P = [p_0, p_1] = [2, 2]$, по аксоновым полям – $G = [g_0, g_1] = [16, 1]$. Результаты классификации входных кодов показаны на рис. 3. Максимальный уровень выхода при классификации ближайшего кода, отличного от собственного не превышает значения 0.886. Компараторы логического нуля на выходах сети должны быть настроены на значения из интервала $[0.886+, 1.000-]$.

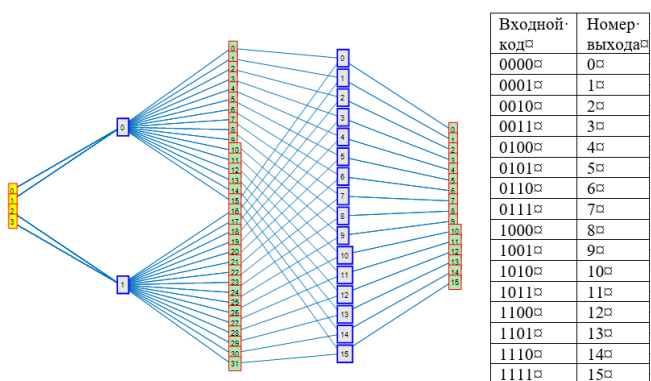


Рис. 2. Дешифратор логических кодов на пирамидальной нейронной сети

Проблемным является входной код [0000]. Скалярное произведение этого кода с эталоном будет равно нулю, в то время как в классическом дешифраторе оно должно равняться единице.

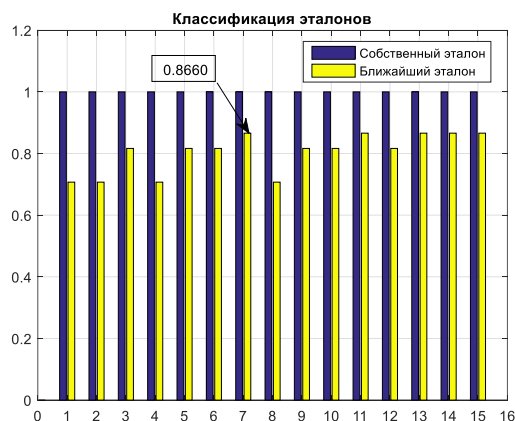


Рис. 3. Классификация логических кодов пирамидальной нейронной сетью

Решением является видоизменение нулевого кода, например, замена одного из нулей кодового слова [0000] значением (-1). Более радикальным вариантом является замена во всех кодовых словах значения (0) на значение (-1), т. е. использование двух-полярного бинарного кода со значениями (-1,+1). Результаты классификации входных кодов для этого случая показаны на рис. 4. Выходные компараторы логического нуля должны быть настроены на уровень 0.5+.

Подобным образом могут быть построены дешифраторы кодов с основанием больше двух.

IV. ЛОГИЧЕСКИЙ ШИФРАТОР

Шифратор выполняет операцию обратную дешифрированию, т. е. преобразует унарный код в кодовые слова. Эту операцию можно выполнить на элементе памяти образов. Рассмотрим, например, построение шифратора для двоичного четыре разрядного кода.

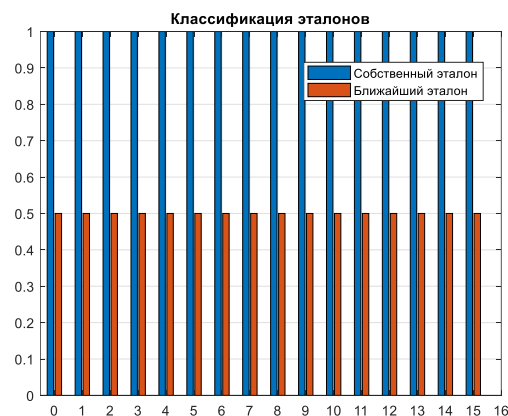


Рис. 4. Классификация двух-полярных бинарных кодов

Разместим в памяти матрицу кодовых слов, тогда логическую функцию шифратора можно реализовать как умножение унарного вектора-строки длиной 16 на матрицу кодовых слов размерностью 16×4 :

$$OUT = [v_3 v_2 v_1 v_0] = [U_0 U_1 U_2 U_3 U_4 U_5 U_6 U_7 U_8 U_9 U_{10} U_{11} U_{12} U_{13} U_{14} U_{15}] \begin{matrix} 0000 \\ 0001 \\ 0010 \\ 0011 \\ 0100 \\ 0101 \\ 0110 \\ 0111 \\ 1000 \\ 1001 \\ 1010 \\ 1011 \\ 1100 \\ 1101 \\ 1111 \end{matrix},$$

где только один элемент вектора-строки равен единице, а остальные равны нулю.

Покажем реализацию памяти образов на основе БНС. Каждую строку матрицы кодовых слов будем рассматривать как функцию $f^k(V) = f^k(v_{n-1}v_{n-2}\dots v_0)$, где k – номер строки, v_i – разрядные переменные. Выполним мультипликативную декомпозицию [7] каждой функции по переменным v_i , начиная со старшего разряда, в результате получим:

$$f^k(V) = \varphi_{i^{n-1}}^k(v_{n-1})\varphi_{i^{n-2}}^k(v_{n-2})\dots\varphi_{i^0}^k(v_0), \quad (6)$$

где $i^m = \langle v_{m-1}v_{m-2}\dots v_1v_0 \rangle$. Выберем структурные характеристики БНС следующим образом:

- размерности рецепторных полей сети положим равными $p_0 = p_1 = \dots = p_{n-2} = 1$, $p_{n-1} \neq 1$. В этом случае размерность сети по входу будет равна $N = p_0 p_1 \dots p_{n-1} = p_{n-1}$;
- размерности аксоновых полей сети зададим произвольными натуральными целыми числами g_0, g_1, \dots, g_{n-1} . Размерность сети по выходу в этом случае будет равна $M = g_0 g_1 \dots g_{n-1}$.

Топологическая модель (2) при данных структурных характеристиках будет иметь вид:

$$\begin{aligned} U^m &= \langle v_{m-1}v_{m-2}\dots v_0 u_{n-1} 0_{n-2} \dots 0_m \rangle, \\ V^m &= \langle v_m v_{m-1} \dots v_0 u_{n-1} 0_{n-2} \dots 0_{m+1} \rangle, \\ z^m &= \langle v_{m-1} \dots v_0 u_{n-1} 0_{n-2} \dots 0_{m+1} \rangle. \end{aligned} \quad (7)$$

Сравнивая выражения (6) и (7) получим правило обучения нейронной сети:

$$\begin{aligned} w_{z^{n-1}}^{n-1}(u_{n-1}, v_{n-1}) &= \varphi_{i^{n-1}}^k(v_{n-1}) \quad \text{для } m = n-1, \\ w_{z^m}^m(0_m, v_m) &= \varphi_{i^m}^k(v_m) \quad \text{для } m < n-1, \\ z^m &= \langle v_{m-1}v_{m-2}\dots v_1v_0 u_{n-1} 0_{n-2} \dots 0_{m+1} \rangle, \\ i^m &= \langle v_{m-1}v_{m-2}\dots v_1v_0 \rangle. \end{aligned}$$

Иначе его можно назвать правилом записи образа в нейро-сетевую память. Предварительно, для корректного упорядочивания хранимых функций, должно быть установлено соответствие $k = u_{n-1}$ между порядковым номером функции и разрядной переменной u_{n-1} . Данная

сеть памяти способна хранить p_{n-1} произвольных образов. Считывание памяти реализуется установкой на входе нейронной сети унарного кода, в котором только один из разрядов равен 1, а остальные равны нулю.

Для построения шифратора 16×4 выберем двухслойную нейронную сеть со структурными характеристиками $P = [p_0, p_1] = [1, 16]$ и $G = [g_0, g_1] = [2, 2]$. Граф топологической модели сети показан на рис. 5.

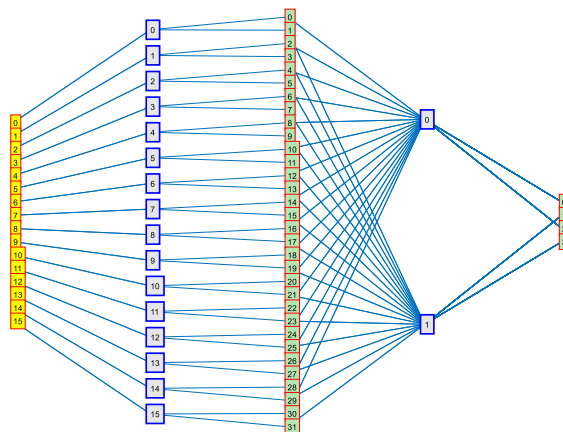


Рис. 5. Нейронная сеть памяти реализующая шифратор 16×4

Двухслойная нейронная сеть по входу имеет размерность 16, а по выходу 4. Ядра нулевого слоя имеют размерности $[1, 2]$, а выходного – $[16, 2]$. Для сети памяти уровни логических значений могут быть стандартными: 0 и 1.

V. ЗАКЛЮЧЕНИЕ

Быстрые нейронные сети относятся к широкому классу самоподобных многослойных нейронных сетей [9]. БНС с регулярной структурой обладают уникальным свойством аналитического представления топологической модели, что позволяет разработать быстрые абсолютно сходящиеся алгоритмы обучения с конечным числом вычислительных операций.

Пирамидальные сети обладают минимально-достаточной структурой для обучения к заданному многообразию функций, можно показать, что их информационная ёмкость используется полностью. Сети данного типа удовлетворяют принципу двойственности, они в равной степени могут быть использованы как для реализации многоканальных корреляторов, так и для создания элементов памяти образов. В данной работе этот принцип продемонстрирован на примере построения шифраторов и дешифраторов бинарных кодов. Подобным образом могут быть реализованы другие сложные логические функции, в том числе функции многозначной логики. Последовательное соединение нейросетевого дешифратора и шифратора можно использовать для построения преобразователей кода, например, равномерного двоичного кода в код Грея. Эта же модель может быть легко перенастроена для выполнения обратного преобразования. Более того, регулярный лес пирамидальных сетей позволяет реализовать быстрые нейронные сети глубокого

обучения с расширенными возможностями по информационной ёмкости [10].

БНС реализующая память способна хранить не только логические функции, но и любые образы, например, в двумерном варианте сети это будут произвольные изображения. Последовательное соединение нейронной сети памяти с дешифратором на БНС, позволяет построить быструю память образов, с адресацией подобной адресации компьютерной памяти произвольного доступа, но элементом хранения здесь будет являться не одиночное число, а образ в целом.

СПИСОК ЛИТЕРАТУРЫ

- [1] Elman, J.L. Finding structure in time. // *Cognitive Science*. 1990. С. 179-211.
- [2] Hopfield J.J. Neural networks and physical systems with emergent collective computational abilities // *Proceedings of National Academy of Sciences*, vol. 79 no. 8 pp. 2554–2558, April 1982.
- [3] Richard Lippmann. 1987. An introduction to computing with neural nets. *IEEE Assp magazine*.
- [4] Carpenter, G.A. & Grossberg, S. (1987), ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics* volume 26, number 23, December, 1987.
- [5] Дорогов А.Ю. Быстрые нейронные сети: Проектирование, настройка, приложения: Лекции по нейронинформатике Ч.1 // В тр. школы-семинара «Современные проблемы нейронинформатики», науч.-техн. конф. “Нейронинформатика-2004” 28-30 января 2004 г. Москва. Изд. М.: МИФИ, 2004, с. 69-135.
- [6] Дорогов А.Ю. Системные инварианты быстрых преобразований // Труды Четвертой межрегиональной школы семинара «Распределенные и кластерные вычисления», 14-16 сентября 2004 года, г. Красноярск.
- [7] Дорогов А.Ю. Теория и проектирование быстрых перестраиваемых преобразований и слабосвязанных нейронных сетей. СПб.: «Политехника», 2014. 328 с.
- [8] Дорогов А.Ю. Корреляционный дискриминатор изображений в классе быстрых нейронных сетей // XXV Международная конференция по мягким вычислениям и измерениям (SCM-2022). Сборник докладов. Санкт-Петербург. 25–27 мая 2022 г. СПб.: СПбГЭТУ «ЛЭТИ», с. 134-137.
- [9] Дорогов А.Ю. Самоподобные структуры многослойных нейронных сетей. // XXII Международная научно-техническая конференция «Нейронинформатика-2020» (10-16 октября 2020г.): Сборник научных трудов. М. НИЯУ МИФИ, 2020, с. 214-224.
- [10] Дорогов А.Ю. Быстрые нейронные сети глубокого обучения. // Сборник докладов III Международной научной конференции по проблемам управления в технических системах (CTS'2019).. Санкт-Петербург. 30 октября - 1 ноября 2019 г. СПб.: СПбГЭТУ «ЛЭТИ». С. 275-280.