

ПРОГРАММНАЯ ИНЖЕНЕРИЯ, ТЕСТИРОВАНИЕ И ВЕРИФИКАЦИЯ ПРОГРАММ

УДК 004.42

НЕКОТОРЫЕ АСПЕКТЫ ПОСТРОЕНИЯ АССОЦИАТИВНОЙ ПАМЯТИ НА НЕЙРОННОЙ СЕТИ ХОПФИЛДА

© 2020 г. Ю. Л. Карпов^{а,*}, Л. Е. Карпов^{б,с,**}, Ю. Г. Сметанин^{д,е,***}

^а ООО Люксфорт Профешнл

123060 Москва, 1-й Волоколамский проезд, 10, Россия

^б Институт системного программирования им. В.П. Иванникова РАН
109004 Москва, ул. Солженицына, 25, Россия

^с Московский Государственный университет имени М.В. Ломоносова
119991 Москва, Ленинские горы, 1, Россия

^д Федеральный исследовательский центр “Информатика и управление” РАН
119333 Москва, ул. Вавилова, д. 44, кор. 2, Россия

^е Московский Физико-Технический институт 141701 Московская область,
г. Долгопрудный, Институтский пер., 9, Россия

*E-mail: y.l.karpov@yandex.ru

**E-mail: mak@ispras.ru

***E-mail: ysmetanin@rambler.ru

Поступила в редакцию 05.03.2020 г.

После доработки 25.03.2020 г.

Принята к публикации 11.04.2020 г.

Уже после сдачи в печать этой статьи безвременно ушел из жизни вдохновитель нашей работы Юрий Геннадьевич Сметанин. Он не оставлял своей работы до самого конца, уже по телефону из больницы обсуждая планы следующих исследований и экспериментов. Светлая память о нашем товарище, настоящим ученом и очень хорошим человеке останется с нами навсегда.

Рассматривается реализация ассоциативной памяти, в основе которой находится нейронная сеть Хопфилда. В предлагаемом подходе адреса в памяти трактуются как обучающие векторы нейронной сети. Эффективность поиска информации в памяти напрямую связана с решением проблемы переобучения. Предлагается метод разделения обучающих и входных векторов сети на части, обработка которых требует меньшего числа нейронов. Описываются результаты серии экспериментов, проводившихся на моделях сетей Хопфилда с различным числом нейронов, обучавшихся с разным количеством векторов и работавших в условиях различного уровня шума.

DOI: 10.31857/S0132347420050027

1. ВВЕДЕНИЕ

Программные модели нейронных сетей представляют собой объекты, исключительно интересные для исследования. С одной стороны, эти модели суть системы программного обеспечения, проектирование, разработка и эксплуатация которых подчиняется всем законам, правилам и стандартам, которым подчиняются другие программные системы. С другой стороны, особенностью этих моделей является то, что в них, как, пожалуй, ни в каких других системах не достигают такой степени важности вопросы структурной организации многочисленных составляющих их компонентов. При том, что каждый отдельный компонент нейронной сети — это достаточно сложно ор-

ганизованный вычислительный комплекс, большое число таких компонентов в сети и разнообразие методов организации их взаимодействия делают проблемы выбора и последующей проверки его правильности чрезвычайно важными и трудными.

Авторы уже привлекали внимание к проблеме организации тестирования структуры нейронных сетей (см. [1–3]).

Мы рассматриваем конкретную модель рекуррентной нейронной сети — модель Хопфилда [см. 4], то есть сеть из n нейронов с полным набором симметричных связей, в которой действуют правила

обновления нейронов — $x_i(t+1) = \text{sgn} \left(\sum_{j=1}^n w_{ij} x_j(t) \right)$,

где n — число нейронов, w_{ij} — вес связи между нейронами i и j , $w_{ji} = w_{ij}$. Доказано, что при асинхронном обновлении нейронов эта сеть всегда сходится к состоянию равновесия, в котором вектор значений нейронов перестает изменяться. Число состояний равновесия называется емкостью сети. При любом начальном состоянии сети она на некотором шаге попадает в одно из состояний равновесия. Для каждого состояния равновесия областью ее притяжения называется множество состояний сети, из которых она в ходе обновления попадает в это состояние. Таким образом, можно считать, что область притяжения — это набор входных векторов с такими искажениями, которые данная нейронная сеть исправляет. Именно поэтому нейронную сеть Хопфилда часто называют ассоциативной памятью. Такая сеть разбивает множество допустимых входных векторов на классы, каждый из которых соответствует своему эталону — состоянию равновесия сети, и распознает принадлежность входного вектора классу.

2. ЕМКОСТЬ, ТОЧНОСТЬ РЕШЕНИЯ, СОСТОЯНИЯ РАВНОВЕСИЯ НЕЙРОННОЙ СЕТИ

Для построения количественных характеристик способности конкретной модели нейронной сети к распознаванию естественно использовать число классов, на которые эта сеть может эффективно разделить множество допустимых векторов. Эта величина называется емкостью данной модели сетей. Емкость является вероятностной характеристикой, поскольку зависит от вида классов, то есть конкретных параметров нейронной сети. Как правило, используются асимптотические значения емкости при $n \rightarrow \infty$. Определения емкости, соответствующие различным моделям и различным применениям, можно найти в работе [5].

Пусть дан эталонный набор из p векторов $x^1, \dots, x^p$ и требуется построить матрицу весов синапсов, при которых каждый из этих векторов будет состоянием равновесия сети Хопфилда. Точное решение этой задачи получается при использовании псевдообратной матрицы для построения матрицы весов,

$$W = X(X^T X)^{-1} X^T$$

где X — матрица размера $n \times p$, столбцами которой являются векторы из данного набора.

Очевидно, при такой матрице все эти векторы X будут состояниями равновесия. Для n векторов в общем положении матрица W является невы-

рожденной, откуда следует, что емкость модели нейронной сети Хопфилда — не менее n .

К сожалению, платой за точность решения с использованием обращений матриц является сложность вычислений обратной матрицы. Поэтому на практике обычно используют более простое правило обучения, позволяющее обойтись без сложных вычислений, то есть полностью использовать достоинства нейронных сетей. Здесь будет рассмотрен распространенный метод обучения с учителем для построения однослойных рекуррентных нейронных сетей Хопфилда — правила Хебба (см. [6]).

Правила Хебба были впервые предложены для объяснения динамики биологических нейронных сетей и выражают принцип “если два нейрона часто загораются и гаснут вместе, связь между ними надо усилить”. Из этого принципа для искусственных нейронных сетей получено простое правило обучения — метод внешнего произведения, который работает следующим образом (для простоты рассмотрим двоичный случай, значения нейронов — $+1$ или -1).

Пусть задан набор векторов $a^1 = (a_i^1), \dots, a^p = (a_i^p)$ и требуется построить матрицу весов синапсов W , для которой эти векторы являются положениями равновесия, $\text{sgn}(W a^k) = a^k$. Для каждого вектора $a^1, \dots, a^p$ строится матрица W^k с элементами $w_{ij}^k = a_i^k a_j^k$. $w_{ij}^k = \begin{cases} a_i^k a_j^k, & i \neq j \\ 0, & i = j \end{cases}$.

$$\text{Очевидно, } \text{sgn}(W^k a^k) = \text{sgn} \left(\sum_{j=1}^n a_i^k a_j^k a_j^k \right) = \text{sgn} a_i^k,$$

то есть a является состоянием равновесия для сети с матрицей W^k . С другой стороны, если вектор b ортогонален a^k , то $(W^k b) = 0$. Следовательно, для ортого-

нальных наборов векторов a^k матрица $W = \sum_{k=1}^p W^k$

обладает требуемым свойством. В общем случае гарантировать этого нельзя, и оценки корректности метода внешнего произведения являются вероятностными.

Статистическая механика модели Хопфилда, впервые проанализированная в [7], основанная на использовании аналогий с моделью спиновых стекол, позволила получить достаточно интересные результаты о работе модели как ассоциативной памяти. Основная идея заключается в анализе ве-

личины $M^\mu = \frac{1}{n} \sum_{i=1}^n s_i^\mu s_i^\mu$, характеризующей сходство произвольного текущего состояния с μ -ым эталоном. Очевидно, что для состояний, не коррелирующих с эталонами, эта величина имеет порядок

$1/\sqrt{N}$, а для состояний, отличающихся от данного эталона в небольшом числе координат, она близка к единице. Число обучающих векторов, на которых можно обучить модель Хопфилда с помощью метода внешнего произведения таким образом, чтобы все они стали состояниями равновесия, в предположении, что обучающие векторы выбраны по схеме Бернулли, — $O(n/\log_2 n)$ (см. [8, 9]). В этом случае, как следует из сказанного раньше, возникают посторонние состояния равновесия.

Вид областей притяжения в сети может быть сложным, эти области даже могут быть несвязными (см. [10]). Вероятностный характер обучения приводит к тому, что области притяжения посторонних состояний равновесия могут искажать области притяжения истинных состояний равновесия. Для ассоциативной сети желательно, чтобы истинные состояния равновесия, соответствующие обучающими эталонам, имели по возможности большие области притяжения, а посторонние были как можно меньше, а в идеале вообще отсутствовали.

Отсюда следует, что эффективность нейронной сети становится малой не только в случае, когда число эталонов слишком велико, и неизбежно такое сходство между ними, которое приводит к искажениям вместо коррекции входных векторов, но и в случае, когда эталонов слишком мало. В последнем случае неизбежно появление больших посторонних областей притяжения, из-за чего трудно оценивать правильность полученных результатов, а кроме того, сохранить существенные различия между обучающими эталонами можно и при их преобразовании в представления меньшей размерности, а, следовательно, появляется возможность снизить размер (сложность) нейронной сети, требуемой для решения поставленной задачи.

Авторами исследовался случай, когда число эталонов значительно меньше критического, и для построения ассоциативной памяти, запоминающей эти эталоны и корректирующей их искажения, оказывалось достаточно значительно меньшей размерности сети. В этом случае естественно использовать метод кодирования со сжатием, однако его реализация требует решения задачи о выборе кода с оптимальным сочетанием размерности кодовых слов и способности к исправлению ошибок. Кроме того, необходимо обеспечить выполнение следующего не очень простого для проверки условия: малые искажения, исправляемые сетью Хопфилда с исходными векторами, останутся исправляемыми и сетью Хопфилда, работающей с закодированными представлениями.

Авторами предложен более простой подход к уменьшению сложности нейронной сети в этом

случае, основанный на разбиении каждого эталона на r частей одинаковой длины с приписыванием каждой части дополнительных вспомогательных компонентов, указывающих соответствующий эталон, позицию части в этом эталоне и, возможно, компонента для выравнивания длин. Обучение сети производится с помощью правил внешнего произведения на новом наборе эталонов, полученным в ходе разбиения исходного набора. При этом вспомогательные компоненты должны точно приписываться и поступающим на вход обученной сети анализируемым векторам, также разбитым на части; эти компоненты не могут изменяться в ходе нейросетевой обработки. По окончании работы нейронной сети исправленные ею части сливаются в единый исправленный вектор.

При этом новое число эталонов $p' = rp$, новая длина векторов $n' = n/r$, а значение r подбирается таким образом, чтобы новое значение $\alpha' = pr^2/n$ было меньше критического.

3. ЭКСПЕРИМЕНТЫ С НЕЙРОННЫМИ СЕТЯМИ

Тестирование немыслимо без эксперимента. Тестирование структуры нейронной сети такой же эмпирический процесс, как и тестирование традиционного программного обеспечения. Несмотря на наличие большого числа теоретических разработок, повышающих отдачу от проведения этого необходимого для создания эффективных, надежных, удобных программ и программных моделей (см. [11–14] и многие другие работы, посвященные тестированию), эксперимент в тестировании не теряет своей важности.

Исследуя возможности влияния на структуру нейронной сети, авторы провели несколько серий экспериментов, позволивших уточнить представления о поведении сетей в различных условиях и сделать важные выводы.

В качестве объекта исследований были выбраны сети Хопфилда, на входы которым подавались одноразрядные значения, трактуемые как числа -1 и $+1$. Количество входов задавалось разным в разных экспериментах и менялось от 13 до 960.

В рамках проводимого исследования нейронная сеть трактовалась нами, как модель ассоциативной памяти, соответственно, все эксперименты планировались нами как попытки запомнить некоторое количество чисел, а затем выполнить серию чтений из памяти при получении “правильных” запросов. “Правильность” запроса определялась как наличие по соответствующему адресу в ассоциативной памяти хранящейся информации. Эксперименты намеренно усложнялись тем, что в качестве запрашиваемых адресов на вход подавались не только те “ассоциативные” адреса, по которым первоначально осуществлялась запись ин-

формации, но и другие, отличающиеся от “правильных” в той или иной степени.

При проведении экспериментов нас особенно интересовала возможность снижения “ширины” сети без потери качества запоминания (или с управляемым снижением этого качества) в случае, когда число запоминаемых объектов невелико по сравнению с априорными оценками емкости сети. Очевидно, что снижение числа нейронов в сети должно приводить к повышению эффективности программных моделей, однако, с аппаратно реализованными нейронными сетями такой простой зависимости не существует. Значительная часть затрат на аппаратную реализацию нейронной сети осуществляется еще на стадии ее проектирования и физического построения. В дальнейшем затраты в основном связаны с энергопотреблением во время работы сети. Очевидно также, что снижение числа нейронов, которого удастся добиться при выборе характеристик сети, необходимых для решения поставленной задачи, выгодно и для аппаратной реализации сетей, и для построения их программных моделей.

Поскольку программная работа с нейронными сетями больших размерностей связана с большими затратами времени, в наших экспериментах с сетями с различным количеством нейронов векторы, подававшиеся на вход сетям, различались не только разрядностью (что очевидно), но и критериями, по которым они генерировались.

Сетям небольшой размерности (в наших экспериментах — это сети из 13 и 23 нейронов) подавались на вход полные наборы всех возможных векторов соответствующих размерностей — 8192 разряда в первом случае и 8388608 во втором. В этих наборах случайным образом авторы выбирали необходимое для проведения эксперимента количество векторов, объявлявшихся эталонными. Именно эти вектора запоминались в ассоциативной памяти, то есть использовались для обучения сетей.

Для более крупных сетей такой подход ввиду невозможности за разумное время обработать большое число входных векторов оказался неприемлемым, поэтому для них был предложен специальный механизм отбора эталонных векторов.

На сайте [15] была выбрана случайная последовательность, составлявшаяся при замерах метеорологических данных в Пекинском международном аэропорту. Некоторые из собранных данных (направление ветра) описывались в этом наборе как азимуты в виде кодов, соответствующих разным географическим направлениям с интервалом в 22,5 градуса, например, направление на север — это *N*, направление на северо-северо-запад — это *NNW*, и так далее. Фактически таким образом кодировались 16 различных направлений, кото-

рые мы стали трактовать как шестнадцать цифр в шестнадцатеричной системе счисления ($N \rightarrow '0'$, $NNE \rightarrow '1'$, ..., $NNW \rightarrow 'f'$). Тем самым, в нашем распоряжении оказалась последовательность из 35064 случайных шестнадцатеричных цифр. Их этих случайных последовательностей были составлены случайные векторы с размерами, соответствовавшими числу нейронов в исследовавшихся нейронных сетях, то есть в 32, 40, 256 и 960 разрядов.

Понятно, что генерация всех возможных входных векторов с соответствующим числом разрядов, хотя и несложна, но очень затратна по времени, поэтому на основе отобранных эталонов мы сгенерировали набор входных векторов, куда включили только те векторы, которые отличались от эталонов не более, чем в одной шестнадцатеричной цифре. При этом для некоторого приближения к более представительной выборке отличия в отдельных шестнадцатеричных цифрах допускались такие, чтобы можно было имитировать одно-, двух-, трех и четырехразрядный шум.

В экспериментах фиксировались следующие параметры сетей:

- длина входного вектора;
- число заданных обучающих (запоминаемых) векторов и верхняя оценка оптимального числа таких векторов;
- уровень шума (максимальное число разрядов, отличающих входные векторы от эталонов);
- число частей, на которые делились входные векторы при построении укороченных сетей;
- число входных векторов, поступивших на вход нейронной сети;
- доля верных решений (число входных векторов, которые были успешно преобразованы к одному из обучающих векторов);
- среднее число шагов, которые были выполнены в сети при обработке одного входного вектора;
- суммарное число шагов, выполненных сетью при обработке всех входных векторов, участвовавших в данном эксперименте.

Во время экспериментов проводились замеры размеров параметров сокращенных входных векторов (размеров информационной и служебных фрагментов векторов), определялась максимальная емкость ассоциативной памяти выбранной размерности, измерялось количество обработанных входных векторов, среднее число шагов, выполненных нейронной сетью для выяснения правильности или ложности полученного решения, либо фиксации нестабильности достигнутого состояния, а также общее число шагов, сделанных при обработке сетью всех входных векторов.

Каждый эксперимент проводился в двух вариантах. Сначала выполнялось моделирование рабо-

Таблица 1. Обучающие векторы в экспериментах с 13-разрядными нейронными сетями

Номер разряда		0	1	2	3	4	5	6	7	8	9	10	11	12
Обучающий вектор 1		-1	1	1	-1	-1	-1	1	1	1	-1	1	1	-1
Обучающий вектор 2		-1	-1	1	1	1	1	-1	1	1	-1	1	1	-1
Части вектора 1	0	-1	-1	-1	1	1	-1	-1	-1	1				
	1	-1	1	-1	-1	1	-1	-1	1	-1				
Части вектора 2	0	1	-1	-1	-1	1	1	1	1	-1				
	1	1	1	1	1	-1	1	1	-1	-1				

Таблица 2. Обучающие векторы в экспериментах с 23-разрядными нейронными сетями

Номера двоичных разрядов		0–2	3–6	7–10	11–14	15–18	19–22
Номер 16-ричного разряда		0	2	3	4	5	6
Обучающий вектор 1		4	4	5	c	b	8
Обучающий вектор 2		2	d	2	7	c	2
Номера двоичных разрядов		0	1	2–5	6–9	10–13	
Части обучающего вектора 1	0	-1	-1	4	4	5	
	1	-1	1	c	b	8	
Части обучающего вектора 2	0	1	-1	2	d	2	
	1	1	1	7	c	2	

ты полной нейронной сети с выбранным числом нейронов. Затем с теми же обучающими и входными векторами проводилась операция деления на части, и моделировалась сокращенная нейронная сеть с уменьшенным числом нейронов. Число частей выбиралось равным двум, лишь для сети из 960 нейронов оно составляло 4.

Каждая часть исходного обучающего вектора строилась из таких полей:

- поле номера обучающего вектора (двоичный логарифм от числа обучающих векторов p , с округлением вверх);
- поле номера части вектора (двоичный логарифм от числа частей r , на которые дробится вектор, с округлением вверх);
- информационное поле (число нейронов n , деленное на число частей, с округлением вверх).

Аналогично строились части входных векторов, при этом поле номера обучающего вектора заполнялось числом, соответствующим номеру обучающего вектора, отличавшегося от входного вектора в наименьшем числе позиций.

В таблице 1 показаны значения элементов обучающих векторов, подававшихся на вход нейрон-

ной сети, состоявшей из 13 нейронов. Результаты экспериментов с этой сетью собраны в таблице 2.

В таблице 3 показаны значения элементов обучающих векторов, подававшихся на вход нейронной сети, состоявшей из 23 нейронов. Эти значения представлены в виде 23-разрядных шестнадцатеричных чисел. Нулевое значение какого-либо двоичного разряда в этих числах соответствует элементу вектора, равного -1 . Единичное значение двоичного разряда числа соответствует единичному элементу вектора. Результаты экспериментов с этой сетью собраны в таблице 2.

Уже по таким небольшим экспериментам с весьма небольшими нейронными сетями стали видны некоторые тенденции. И главное, что обратило на себя внимание, это сильная зависимость укороченных сетей от шума в поступающих на вход векторах. Даже относительно небольшие искажения во входной информации приводят к резким снижениям доли правильно распознанных векторов. Чтобы выявить другие проблемы и условия их возникновения, потребовалось провести несколько дополнительных экспериментов с сетями существенно большего размера. Результаты этих экспериментов показаны в таблицах 4 и 5.

Таблица 3. Результаты эксперимента с малоразмерными нейронными сетями

Длина входного вектора	13				23			
Число эталонов	2				2			
Шум (разрядов)	0		4		0		4	
Число частей		2		2		2		2
Информационная часть		7		7		12		12
Размер области номера эталона		1		1		1		1
Размер области номера части		1		1		1		1
Размер обрабатываемого вектора	13	9	13	9	23	14	23	14
Максимальная емкость памяти	4	3	4	3	6	4	6	4
Число обучающих векторов	2	4	2	4	2	4	2	4
Число входных векторов	8192				8388608			
Число обработанных векторов	2		1360		2		17710	
Доля верных решений (процентов)	100	100	78.23	5	100	50	99.21	19.09
Среднее число шагов сети	1	1	2.765	3.028	1	2	2	2.678
Полное число шагов сети	2	4	3760	8236	2	8	35420	94867

4. ВЫВОДЫ

Проведенные эксперименты позволили сделать некоторые выводы. У укороченных сетей более высокие требования к подбору обучающих векторов, при предлагаемом разделении могут возникать векторы с большей корреляцией, то есть увеличивается риск ложных ассоциаций.

Укороченные сети весьма чувствительны к количеству обучающих векторов, точнее при выборе обучающих векторов очень важно представлять себе, что их количество пропорционально возрас-

тает с увеличением количества частей, на которые должны делиться обучающие и входные векторы. Пренебрежение этим может привести к проявлению проблем переобучения нейронной сети.

Укороченные сети более чувствительны к шумам, то есть при росте помех при передаче данных на вход нейронной сети может ухудшаться способность ассоциативной памяти исправлять некоторые погрешности, которые могут возникать при передаче информации в сеть на обработку.

Таблица 4. Результаты эксперимента с нейронными сетями

Длина входного вектора	32				40			
Число эталонов	3				3			
Шум (разрядов)	0		4		0		4	
Число частей		2		2		2		2
Информационная часть		16		7		20		20
Размер области номера эталона		2		1		2		2
Размер области номера части		1		1		1		1
Размер обрабатываемого вектора	32	19	32	9	40	23	40	23
Максимальная емкость памяти	7	5	7	5	8	6	8	6
Число обучающих векторов	3	6	3	6	3	6	3	6
Число обработанных векторов	841		10		811		12	
Доля верных решений (процентов)	100	0	100	0	100	100	100	41.67
Среднее число шагов сети	1	2.5	2	3.1	1	1	2	2.333
Полное число шагов сети	841	4205	20	62	811	1662	24	56
Время работы сети (тактов)	180	180	173	173	210	211	209	206

Таблица 5. Результаты эксперимента с нейронными сетями

Длина входного вектора	256				960			
Число эталонов	6				20			
Шум (разрядов)	0		4		0		4	
Число частей		2		2		4		4
Информационная часть		128		128		240		240
Размер области номера эталона		3		3		5		5
Размер области номера части		1		1		2		2
Размер обрабатываемого вектора	256	132	256	132	960	247	960	247
Максимальная емкость памяти	32	19	32	19	97	32	97	32
Число обучающих векторов	6	12	6	12	20	80	20	80
Число обработанных векторов	116		367		81020		4800	
Доля верных решений (процентов)	100	100	100	89.65	100	0	100	0
Среднее число шагов сети	1	1	2	1.631	1.4	13.875	2.27	13.89
Полное число шагов сети	116	232	367	1197	113428	4496610	10880	266751

В аппаратных реализациях нейронных сетей укорачиванием сети достигается экономия оборудования (но вряд ли экономится время решения задачи). В программных реализациях нейронных сетей укорачиванием сети достигается экономия времени решения задачи (хотя чуть меньшая, чем прямая пропорция с количеством частей из-за включения в укороченные входные векторы двух служебных полей). Хотя число шагов сети при уменьшении числа нейронов в ней возрастает, но каждый такой шаг выполняется с вовлечением в процесс гораздо меньшего числа нейронов. При моделировании операций умножения матрицы на вектор число выполняемых операций при этом уменьшается пропорционально второй степени числа частей, а это приводит к почти линейному сокращению времени моделирования.

В дальнейшем целесообразно провести исследование других способов входных представлений, наилучшим образом сочетающих требования к сложности сети и возможности ассоциативной памяти.

5. БЛАГОДАРНОСТИ

Авторы выражают благодарность Российскому фонду фундаментальных исследований, который поддержал эту работу своими грантами (проекты 18-07-00697-а, 18-07-01211-а, 19-07-00321-а и 19-07-00493-а).

СПИСОК ЛИТЕРАТУРЫ

1. Карпов Ю.Л., Карпов Л.Е., Сметанин Ю.Г. Адаптация общих концепций тестирования программно-обеспечения к нейронным сетям. Программирование. 2018. т. 44. № 5. С. 43–56. DOI: / Yu.L. Karpov, L.E. Karpov, Yu.G. Smetanin. Adaptation of General Concepts of Software Testing to Neural Networks. Programming and Computer Software. 2018. V. 44. № 5. P. 324–334. DOI: <https://doi.org/10.1134/S0361768818050031>. <https://doi.org/10.31857/S013234740001214-0>
2. Карпов Ю.Л., Карпов Л.Е., Сметанин Ю.Г. Устранение отрицательных циклов в некоторых структурах нейронных сетей с целью достижения стационарных решений. Программирование. 2019. Т. 45. № 5. С. 25–35. DOI: / Yu.L. Karpov, L.E. Karpov, Yu.G. Smetanin. Elimination of Negative Circuits in Certain Neural Network Structures to Achieve Stable Solutions. Programming and Computer Software. 2019. V. 45. № 5. P. 241–250. DOI: <https://doi.org/10.1134/S0361768819050025>. <https://doi.org/10.1134/S0132347419050029>
3. Карпов Ю.Л., Волкова И.А., Вылиток А.А., Карпов Л.Е., Сметанин Ю.Г. Проектирование интерфейсов классов графовой модели нейронной сети. Труды ИСП РАН. 2019. Т. 31. Вып. 4. С. 97–112. / Karpov Yu.L., Volkova I.A., Vylitok A.A., Karpov L.E., Smetanin Yu.G. Designing classes' interfaces for neuron network graph model. Programming and Computer Software. 2020. V. 46, to be published. [https://doi.org/10.15514/ISPRAS-2019-31\(4\)-6](https://doi.org/10.15514/ISPRAS-2019-31(4)-6)
4. Hopfield J.J. Neural networks and physical systems with emergent collective computational properties. Proc. Nat. Acad. Sci. USA. 1982. V. 79. P. 2554–2558.
5. Smetanin Yu.G. Neural networks as systems for recognizing patterns. In: Journal of Mathematical Sciences. 1998. V. 89. № 4. P. 1406–1457.
6. Hebb D.O. The Organization of Behavior. Wiley, 1948.
7. Amit D.J., Gutfreund H., Sompolinski H. Spin-Glass Models of Neural Networks, Phys. Rev. A. 1985. V. 32. P. 1007.
8. McEliece R.J., Posner E.G., Rodemich E.R., Venkatesh S.S. The Capacity of the Hopfield Associative Memory, IEEE Trans. Inf. Theory. 1987. V. 33. № 4. P. 461–482.
9. Venkatesh S.S. Computation and Learning in the Context of Neural Network Capacity, Neural Networks for Perceptions, Wechsler, H., Ed., Academic; Harcourt Brace Jovanovich. 1992. V. 2. P. 173–207.

10. *Makhoul J., El-Jaroudi A., Schwartz R.* Formation of Disconnected Regions with a Single Hidden Layer, Int. Joint Conf. on Neural Networks, Washington, DC. 1989. V. 1. P. 455–460.
11. *Майерс Г.* Искусство тестирования программ, М., Финансы и статистика, 1982 / *Myers, G. J.*, The Art of Software Testing, John Wiley & Sons, 1979.
12. *Канер С., Фолк Д., Енг Кек Нгуен.* Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений, Киев, Диасофт, 2001 / *Cem Kaner, Jack Falk, Hung Quoc Nguyen*, Testing Computer Software, Second Edition, International Thompson Publishing Press, 1999.
13. *Тампе Л.* Введение в тестирование программного обеспечения, М.: Издательский дом “Вильямс”, 2003 / *Louise Tamres*, Introducing Software Testing, Addison Wesley, 2002.
14. *Бейзер Б.* Тестирование черного ящика. Технологии функционального тестирования программного обеспечения и систем, Питер, 2004 / *B. Beizer*, “Software Testing Techniques,” Second Edition, Van Nostrand Reinhold Company Limited, 1990.
15. <https://archive.ics.uci.edu/ml/datasets/Beijing+PM2.5+Data#>