

**ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И ПРИНЯТИЕ РЕШЕНИЙ  
ARTIFICIAL INTELLIGENCE AND DECISION MAKING**

УДК 004.855.5

DOI: 10.18413/2518-1092-2022-8-1-0-8

**Черных В.С.  
Жихарев А.Г.  
Федосеев А.Д.  
Мартон Н.А.****СРАВНЕНИЕ ЭФФЕКТИВНОСТИ РАЗЛИЧНЫХ МЕТОДОВ  
ОБУЧЕНИЯ НЕЙРОННЫХ СЕТЕЙ**Белгородский государственный технологический университет им. В.Г. Шухова,  
ул. Костюкова, д. 46, Белгород, 308012, Россия*e-mail: zhikharev@bsu.edu.ru***Аннотация**

В работе рассматриваются несколько подходов к обучению многослойных полносвязных нейронных сетей. В частности, авторами разработана искусственная нейронная сеть, целью которой является распознавание изображений цифр от нуля до шести. Для обучения нейронной сети сформирована обучающая выборка. Авторами проведены ряд экспериментов по реализации различных методов обучения рассматриваемой искусственной нейронной сети. Приводится описание процедуры обучения сети с использованием классического генетического алгоритма. Результаты показали, что генетический алгоритм в классической форме малоэффективен для решения поставленной задачи, так как время обучения искусственной нейронной сети существенно выше в сравнении с алгоритмом обратного распространения ошибки. Также был предложен комбинированный метод обучения, основанный на генетическом алгоритме и градиентном спуске. Результаты эксперимента показали близкие результаты по эффективности в сравнении с алгоритмом обратного распространения ошибки. Из этого следует, что генетический алгоритм применим для решения задач обучения искусственных нейронных сетей.

**Ключевые слова:** нейронная сеть; обучение нейронной сети; полносвязная нейронная сеть; метод; генетические алгоритмы; методы оптимизации; алгоритм обратного распространения ошибки; градиентный спуск; особь; нейрон

**Для цитирования:** Черных В.С., Жихарев А.Г., Федосеев А.Д., Мартон Н.А. Сравнение эффективности различных методов обучения нейронных сетей // Научный результат. Информационные технологии. – Т.8, №1, 2023. – С. 83-93. DOI: 10.18413/2518-1092-2022-8-1-0-8

**Chernykh V.S.  
Zhikharev A.G.  
Fedoseev A.D.  
Marton N.A.****COMPARISON OF EFFICIENCY OF DIFFERENT METHODS  
OF TRAINING NEURAL NETWORKS**Belgorod State Technological University named after V.G. Shukhov,  
46 Kostyukova st., Belgorod, 308012, Russia*e-mail: zhikharev@bsu.edu.ru***Abstract**

The paper considers several approaches to training multilayer fully connected neural networks. In particular, the authors have developed an artificial neural network, the purpose of which is to recognize images of numbers from zero to six. To train the neural network, a training set was formed. The authors carried out a number of experiments on the implementation of various learning

methods for the considered artificial neural network. The description of the network training procedure using the classical genetic algorithm is given. The results showed that the genetic algorithm in the classical form is ineffective for solving the problem, since the training time of the artificial neural network is significantly higher compared to the backpropagation algorithm. A combined learning method based on a genetic algorithm and gradient descent has also been proposed. The results of the experiment showed close results in terms of efficiency in comparison with the backpropagation algorithm. It follows from this that the genetic algorithm is applicable for solving the problems of training artificial neural networks.

**Keywords:** neural network; neural network training; fully connected neural network; method; genetic algorithms; optimization methods; error backpropagation algorithm; gradient descent; individual; neuron

**For citation:** Chernykh V.S., Zhikharev A.G., Fedoseev A.D., Marton N.A. Comparison of efficiency of different methods of training neural networks // Research result. Information technologies. – Т.8, №1, 2023. – P. 83-93. DOI: 10.18413/2518-1092-2022-8-1-0-8

## **ВВЕДЕНИЕ**

Нейронная сеть — это метод в искусственном интеллекте, который учит компьютеры обрабатывать данные таким же способом, как и человеческий мозг. В настоящее время нейронные сети применяются во многих сферах человеческой жизни, от распознавания продуктов в супермаркете до имитации настоящего человеческого сознания. Естественно, с популяризацией нейронных сетей, появилось множество методов обучения [1-3].

Метод обратного распространения ошибки [4] является одним из основных способов обучения и содержит в своей основе алгоритм вычисления градиентного спуска. Другими словами, двигаясь вдоль градиента, происходит расчет локального максимума и минимума функции.

Идея алгоритма обратного распространения ошибки состоит в том, чтобы на основе расчетной ошибки, полученной на выходном слое нейронной сети, пересчитать значение весов вектора  $W$  последнего слоя нейронов. Затем мы переходим к предыдущему слою и так далее, от конца к началу, то есть, он состоит из обновления всех весов  $W$  слоев, от последнего до достижения входного слоя сети путем обратного распространения ошибки, полученной сетью.

Еще один распространенный подход – это обучение нейронной сети генетическим алгоритмом. По своему принципу он схож с эволюционными процессами природы, которые основываются на комбинировании (скрещивании) результатов [5].

Другими словами, происходит естественный отбор, где новое поколение является продуктом комбинации результатов с самыми лучшими свойствами. Если итог такого скрещивания не подходит по каким-то критериям, то отбор совершается вновь, пока продукт не станет совершенным.

Существуют два типа обучения искусственных нейронных сетей: с учителем и без учителя. Обучение нейронной сети можно сравнить с обучением человека. Человек может обучаться на собственном опыте или с учителем, который покажет ему верный ответ. Обучение «с учителем» применяется в тех случаях, когда мы хотим обучить сеть ограниченной выборке классов [6].

При обучении с учителем все примеры обучающей выборки содержат правильные ответы (выходы), соответствующие исходным данным (входам). В процессе обучения веса (коэффициенты) настраиваются так, чтобы сеть порождала ответы, максимально близкие к правильным [7].

## **НАЧАЛЬНЫЕ УСЛОВИЯ**

В данной работе будут рассмотрены 3 метода обучения нейронной сети для решения задачи классификации изображений: метод обратного распространения ошибки, генетический алгоритм и их комбинация.

В качестве обучающего набора данных выступают изображения цифр от 0 до 6 размером 10x20, в двух вариациях, целые и искажённые, то есть с недостающими пикселями, примеры которых показаны на рисунке 1.



Рис. 1. Обучающий набор цифр от 0 до 6

Fig. 1. Training set of digits from 0 to 6

Нейронная сеть в данном примере является полносвязной и состоит из 3 слоёв: входной, скрытый и выходной. Во входном слое 200 нейронов (10x20), в скрытом 100, в выходном 7 (7 цифр).

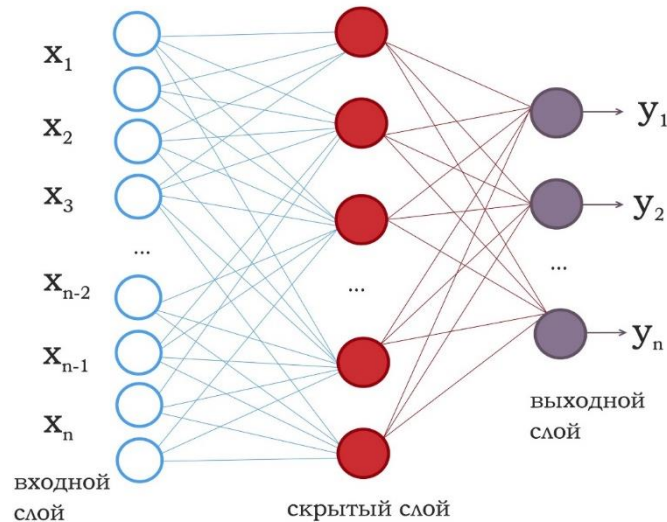


Рис. 2. Полносвязная нейронная сеть

Fig. 2. Fully connected neural network

На входной слой подаётся массив из 200 цифр, состоящих только из 0 и 1 (1 – чёрный, 0 – белый). Их порядок соответствует расположению черных и белых пикселей на картинке. Начальные веса связей между нейронами задаются в диапазоне от -0.5 до 0.5 в случайном порядке. Значения на нейронах выходного слоя означают процентную вероятность принадлежности поданной на вход картинке к тому или иному классу.

### МЕТОДЫ ОБУЧЕНИЯ

Генетический алгоритм представляет собой технику оптимизации, которая моделирует феномен естественной эволюции. При описании генетических алгоритмов используются определения, заимствованные из генетики. Например, речь идет о популяции особей, а в качестве базовых понятий применяются ген, хромосома, генотип, фенотип, аллель.

Популяция – это конечное множество особей (в данной работе размер популяции составляет 200 особей). Особи, входящие в популяцию, в генетических алгоритмах представляются хромосомами с закодированным в них множествами параметров задачи, т.е. решений, которые иначе называются точками в пространстве поиска. Генетический алгоритм не пытается оптимизировать единственное решение. Он работает с группой решений, которые кодируются, подобно хромосомам. Отдельные гены хромосомы представляют собой уникальные переменные для изучаемой проблемы. Алгоритм работы основного (классического) генетического алгоритма представлен на рисунке ниже.

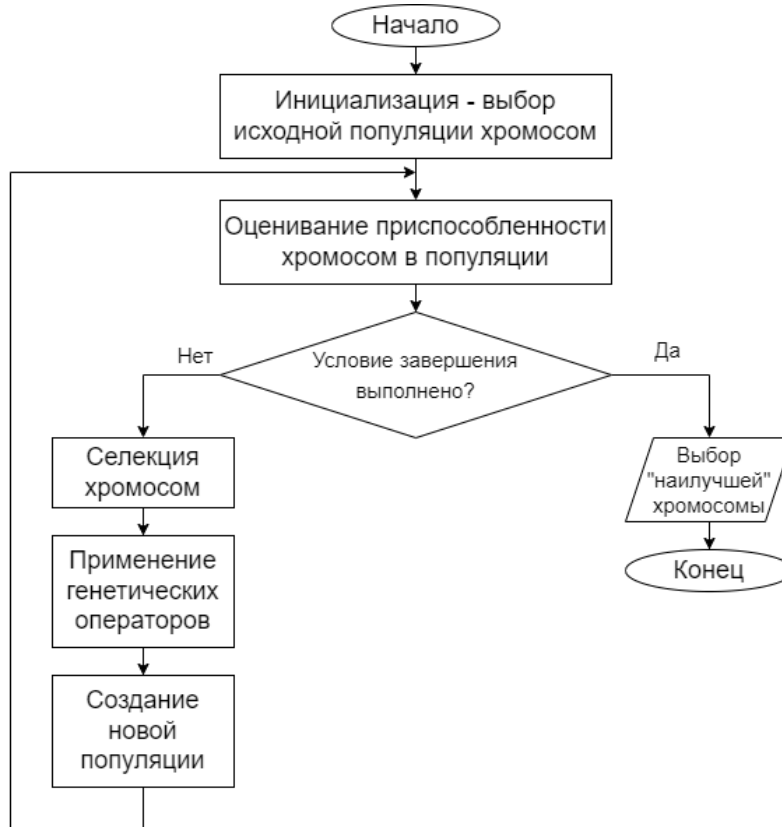


Рис. 3. Блок-схема генетического алгоритма  
Fig. 3. Flowchart of the Genetic Algorithm

Рассмотрим подробнее этапы генетического алгоритма. Создание начальной популяции позволяет сформировать отправную точку для работы алгоритма. Обычно это выполняется путем произвольного создания хромосом. Этап оценки дает возможность определить, как каждая хромосома (решение) справляется с данной проблемой. Алгоритм декодирует хромосому применительно к проблеме и проверяет результат решения проблемы с использованием новых параметров. Затем на основании результата рассчитывается степень приспособленности хромосомы.

На каждой итерации каждой особи даётся абстрактная оценка её приспособленности, показывающая насколько хорошо она подходит к рассматриваемой задаче. Этот метод оценивания будет использоваться в дальнейшем в остальных методах, рассматриваемых в этой работе. Для нахождения наиболее приспособленной особи, у которой верный нейрон приближен к единице, а остальные стремятся к нулю, используются следующие правила накопления очков:

1) Особи с наименьшим средним значением получают больше очков, нежели другие  $\text{score} += 1 - \text{average}$ .

2) Особи, у которых верный нейрон выше всех остальных получает дополнительные очки  $\text{score} += \text{up} / \text{count}$ .

3) Особи, у которых наибольшая большая разница между верным нейроном и максимальным неверным в пользу верного, получает дополнительные очки, в обратном случае теряет их.

4) Особи, преодолевшие границу в 50% приспособленности, получают дополнительные очки, в обратном случае теряют их  $\text{score} += \text{ans} - 0.5$ .

На этапе отбора хромосомы выбираются для дальнейшего использования в другой популяции. Отбор осуществляется на основании степени приспособленности хромосом. Метод отбора по правилу рулетки заключается в случайном выборе особи из популяции исходя из его степени приспособленности. Предположим, у нас имеется несколько особей в популяции с разной степенью адаптации. Тогда их можно условно представить на круговой диаграмме с секторами, размером

соответствующих долей. Затем, мы раскручиваем этот круг и тот сектор, на который будет указывать стрелка, будет выбран.

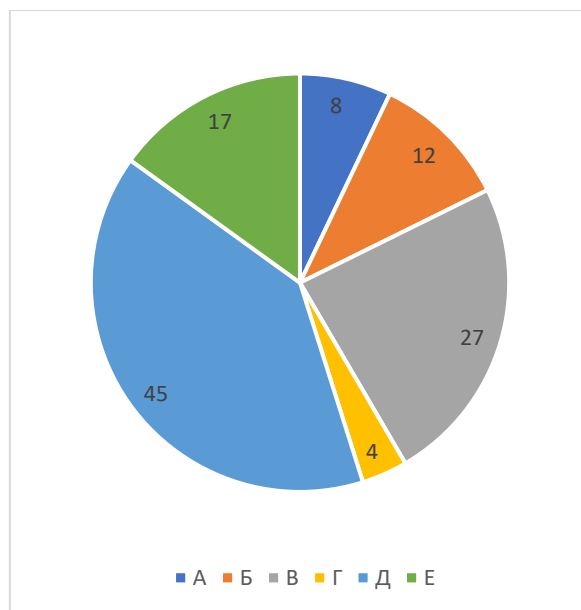


Рис. 4. Схема отбора методом рулетки  
Fig. 4. Roulette selection scheme

При рекомбинировании части хромосом перемещаются, может быть, даже изменяются, а получившиеся новые хромосомы возвращаются обратно в популяцию для формирования следующего поколения. Первая группа хромосом обычно называется родителями, а вторая – детьми. С одинаковой вероятностью могут применяться один или несколько генетических операторов.

Рассмотрим подробнее операции скрещивания. В самом простом варианте операция кроссинговера выполняет обмен между двумя половинками хромосом родителей для формирования хромосом потомков.

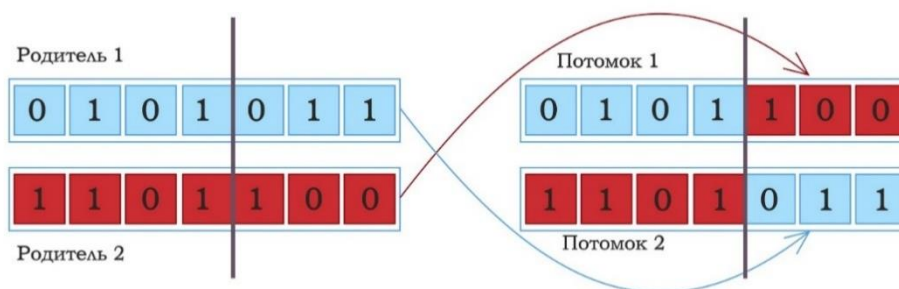


Рис. 5. Одноточечное скрещивание  
Fig. 5. Single point crossing

Вначале случайным образом определяется точка разреза хромосомы, а затем, соответствующие части меняются местами. Получаются две новые хромосомы для двух потомков.

На этапе мутации выбираются случайные гены (веса между нейронами) и к ним прибавляется случайное число от -0.5 до 0.5, если же число стало больше 0.5, то вычитается 1, если меньше -0.5, то прибавляется 1 (в данной работе вероятность мутации потомков составляет 15%).

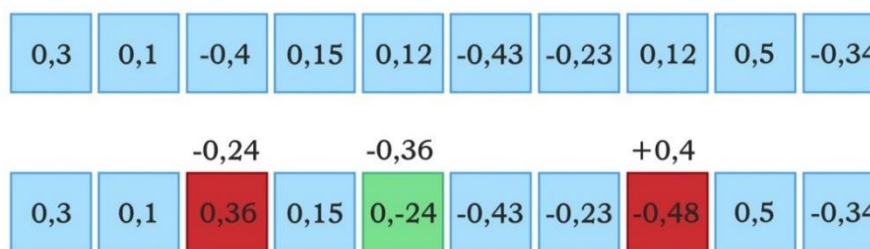


Рис. 6. Схема оператора мутации  
Fig. 6. Scheme of the mutation operator

Далее рассмотрим подробнее метод обратного распространения ошибки. Обратное распространение – это самый популярный алгоритм для обучения с помощью изменения весов связей. Как ясно из названия, ошибка распространяется от выходного слоя к входному, то есть в направлении, противоположном направлению прохождения сигнала при нормальном функционировании сети. На эффективность алгоритма влияет параметр коэффициента обучения, который подбирается эмпирически (в данной работе параметр коэффициента обучения равен 0.3).

Выполнение алгоритма начинается с создания произвольно сгенерированных весов для многослойной сети. Затем процесс, описанный ниже, повторяется до тех пор, пока средняя ошибка на входе не будет признана достаточно малой:

1. Берется пример входного сигнала с соответствующим правильным значением выхода.
2. Рассчитывается прямое распространение входной сигнал через сеть.
3. Начиная с выходов, выполняется обратное движение через ячейки выходного и промежуточного слоя с расчётом значений ошибок.
4. Веса в сети корректируются относительно рассчитанных значений.

### РЕЗУЛЬТАТЫ РАБОТ АЛГОРИТМОВ

Авторами была проведена попытка смешать оба вышеперечисленных метода, за основу был взят генетический алгоритм, но вместо мутации использовался метод обратного распространения ошибки, вероятность мутации потомков была увеличена до 45%, а размер популяции был уменьшен до 50 особей. Результаты обучения искусственной нейронной сети представлены ниже. Рассмотрим результаты обучения с использованием генетического алгоритма.

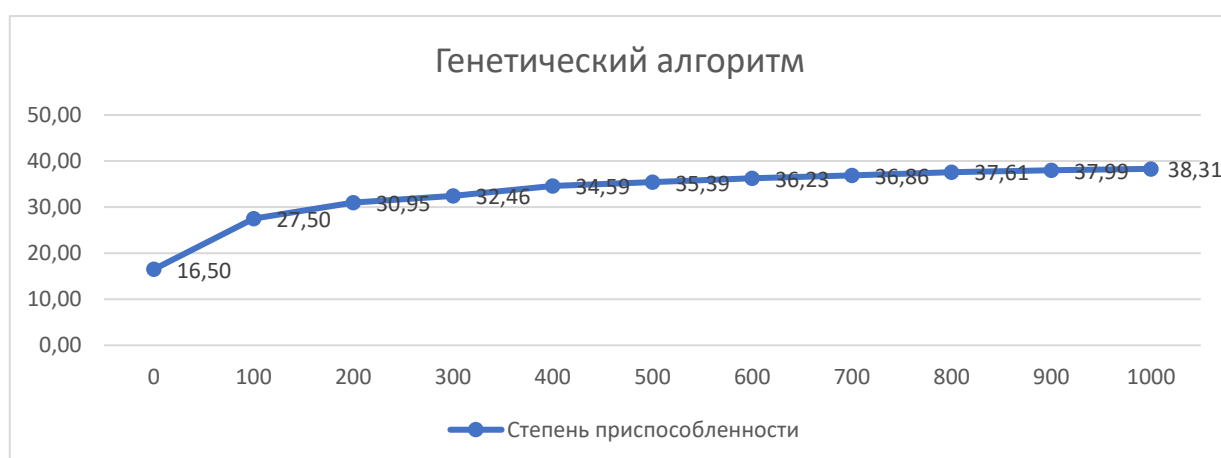


Рис. 7. График изменения степени приспособленности  
Fig. 7. Plot of the degree of fitness

Таблица 1

Лучшая особь первого поколения из генетического алгоритма

Table 1

The best individual of the first generation from the genetic algorithm

	0	1	2	3	4	5	6
0	<b>50%</b>	61%	21%	38%	16%	17%	67%
0(искажённая)	<b>58%</b>	52%	20%	44%	15%	27%	67%
1	17%	<b>87%</b>	52%	40%	69%	47%	26%
1(искажённая)	20%	<b>82%</b>	36%	29%	69%	30%	18%
2	30%	87%	<b>69%</b>	45%	49%	26%	84%
2(искажённая)	34%	87%	<b>73%</b>	49%	41%	26%	78%
3	30%	57%	52%	<b>44%</b>	22%	24%	68%
3(искажённая)	18%	67%	42%	<b>40%</b>	33%	26%	71%
4	27%	76%	59%	39%	<b>81%</b>	29%	40%
4(искажённая)	19%	90%	63%	28%	<b>78%</b>	32%	42%
5	26%	61%	42%	42%	26%	<b>48%</b>	42%
5(искажённая)	30%	70%	46%	56%	27%	<b>42%</b>	51%
6	50%	69%	48%	38%	33%	55%	<b>64%</b>
6(искажённая)	32%	76%	49%	54%	35%	44%	<b>53%</b>

Таблица 2

Лучшая особь тысячного поколения из генетического алгоритма

Table 2

The best individual of the thousandth generation from the genetic algorithm

	0	1	2	3	4	5	6
0	<b>86%</b>	11%	8%	6%	12%	13%	21%
0(искажённая)	<b>87%</b>	9%	7%	6%	11%	11%	12%
1	4%	<b>91%</b>	21%	14%	22%	16%	8%
1(искажённая)	5%	<b>93%</b>	19%	15%	23%	16%	8%
2	14%	13%	<b>95%</b>	10%	13%	6%	11%
2(искажённая)	9%	15%	<b>94%</b>	15%	14%	9%	13%
3	23%	14%	8%	<b>86%</b>	6%	12%	16%
3(искажённая)	17%	18%	12%	<b>90%</b>	6%	14%	15%
4	4%	15%	11%	9%	<b>94%</b>	9%	11%
4(искажённая)	10%	16%	15%	13%	<b>93%</b>	9%	15%
5	12%	7%	4%	14%	9%	<b>92%</b>	13%
5(искажённая)	13%	6%	2%	12%	9%	<b>94%</b>	15%
6	33%	15%	15%	11%	15%	32%	<b>80%</b>
6(искажённая)	18%	13%	15%	9%	20%	21%	<b>86%</b>

Время вычисления одного поколения составляет в среднем 5.5 секунд. Далее рассмотрим результаты обучения искусственной нейронной сети с использованием алгоритма обратного распространения ошибки.

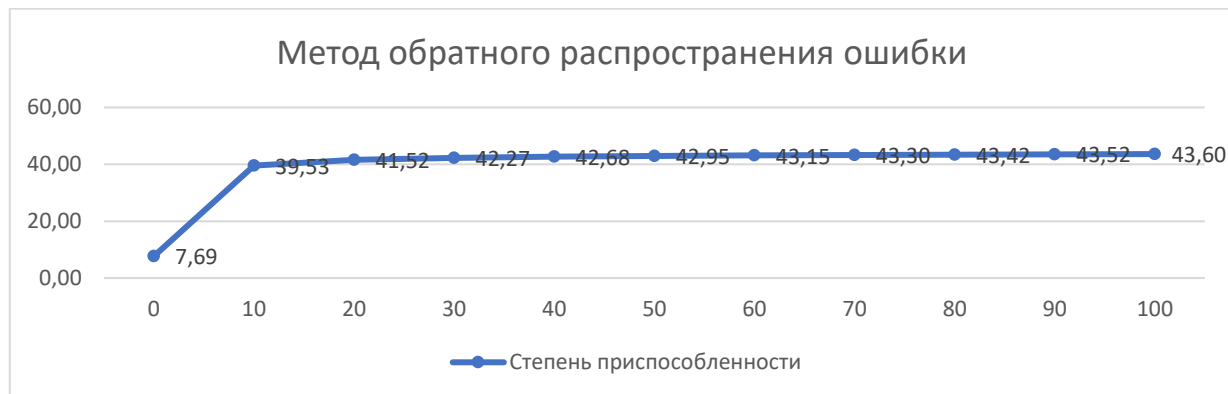


Рис. 8. График изменения степени приспособленности с использованием градиентного спуска  
Fig. 8. Plot of the degree of fitness using gradient descent

Таблица 3

Результат работы нейронной сети в первой итерации в методе обратного распространения ошибки

Table 3

The result of the work of the neural network in the first iteration in the backpropagation method

	0	1	2	3	4	5	6
0	<b>9%</b>	14%	93%	43%	97%	26%	93%
0(искажённая)	<b>28%</b>	10%	81%	16%	97%	14%	80%
1	51%	<b>18%</b>	47%	16%	77%	8%	39%
1(искажённая)	19%	<b>56%</b>	19%	10%	29%	7%	21%
2	22%	81%	<b>10%</b>	26%	58%	3%	25%
2(искажённая)	15%	33%	<b>31%</b>	9%	8%	5%	12%
3	20%	5%	65%	<b>8%</b>	16%	27%	4%
3(искажённая)	11%	7%	19%	<b>24%</b>	13%	8%	6%
4	5%	11%	34%	45%	<b>6%</b>	1%	11%
4(искажённая)	5%	16%	18%	22%	<b>16%</b>	2%	8%
5	9%	2%	8%	53%	60%	<b>6%</b>	19%
5(искажённая)	14%	3%	5%	19%	5%	<b>12%</b>	8%
6	7%	2%	16%	3%	29%	45%	<b>4%</b>
6(искажённая)	5%	2%	10%	1%	15%	10%	<b>14%</b>

Таблица 4

Результат работы нейронной сети в сотой итерации в методе обратного распространения ошибки

Table 4

The result of the work of the neural network in the hundredth iteration in the backpropagation method

	0	1	2	3	4	5	6
0	<b>96%</b>	1%	2%	2%	1%	2%	3%
0(искажённая)	<b>96%</b>	1%	1%	2%	1%	2%	1%
1	1%	<b>97%</b>	1%	2%	2%	2%	0%
1(искажённая)	1%	<b>97%</b>	2%	2%	2%	2%	1%
2	1%	1%	<b>97%</b>	1%	2%	0%	1%
2(искажённая)	1%	1%	<b>97%</b>	2%	1%	1%	1%
3	2%	1%	1%	<b>96%</b>	1%	2%	2%
3(искажённая)	2%	2%	1%	<b>97%</b>	2%	2%	2%
4	0%	2%	1%	1%	<b>96%</b>	1%	2%
4(искажённая)	1%	2%	1%	1%	<b>97%</b>	1%	2%
5	1%	1%	1%	1%	0%	<b>97%</b>	2%



5(искажённая)	2%	1%	1%	2%	0%	<b>97%</b>	2%
6	4%	0%	2%	2%	1%	1%	<b>95%</b>
6(искажённая)	2%	1%	1%	1%	2%	2%	<b>97%</b>

Время вычисления одной итерации составляет в среднем 0.15 секунды, что гораздо эффективнее при использовании классического генетического алгоритма. Далее рассмотрим результаты обучения искусственной нейронной сети смешанным методом.

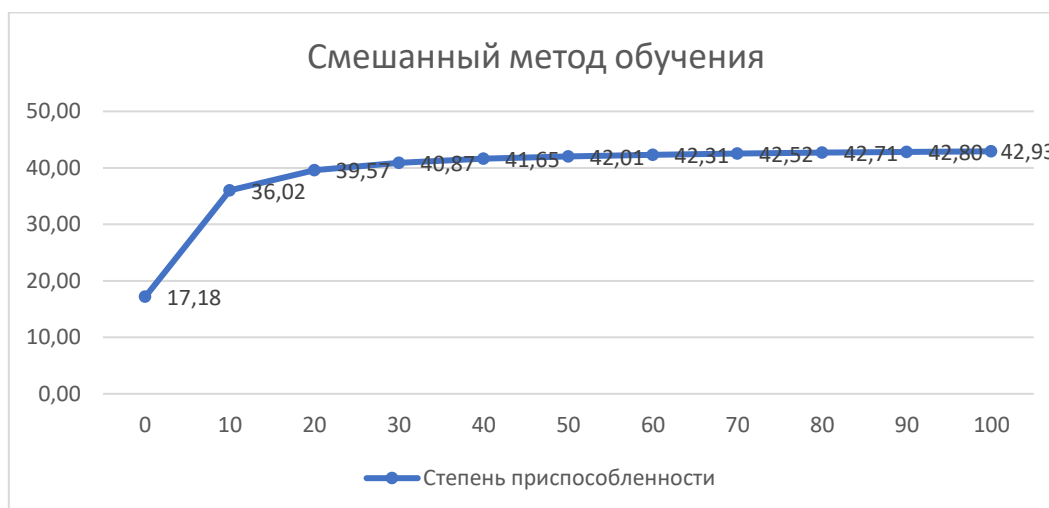


Рис. 9. График изменения степени приспособленности при использовании смешанного метода обучения

Fig. 9. Plot of changes in the degree of fitness when using a mixed learning method

Таблица 5

Лучшая особь первого поколения из смешанного метода обучения

Table 5

The best individual of the first generation from blended learning method

	0	1	2	3	4	5	6
0	<b>41%</b>	31%	9%	59%	38%	67%	39%
0(искажённая)	<b>40%</b>	34%	13%	42%	28%	74%	27%
1	11%	<b>56%</b>	63%	40%	88%	86%	23%
1(искажённая)	15%	<b>55%</b>	68%	41%	87%	85%	12%
2	42%	55%	<b>28%</b>	59%	78%	68%	51%
2(искажённая)	33%	50%	<b>32%</b>	64%	68%	73%	58%
3	13%	49%	45%	<b>59%</b>	34%	42%	34%
3(искажённая)	15%	44%	37%	<b>53%</b>	45%	45%	28%
4	13%	54%	13%	27%	<b>87%</b>	59%	37%
4(искажённая)	14%	42%	20%	27%	<b>91%</b>	54%	31%
5	50%	19%	42%	21%	22%	<b>71%</b>	52%
5(искажённая)	39%	27%	47%	19%	29%	<b>68%</b>	44%
6	30%	31%	10%	63%	39%	43%	<b>66%</b>
6(искажённая)	36%	30%	13%	60%	30%	56%	<b>72%</b>

Таблица 6

Лучшая особь сотого поколения из смешанного метода обучения

Table 6

The best individual of the hundredth generation from blended learning method

	0	1	2	3	4	5	6
0	<b>94%</b>	2%	2%	2%	0%	1%	4%
0(искажённая)	<b>94%</b>	1%	1%	2%	0%	2%	3%
1	2%	<b>95%</b>	4%	2%	4%	2%	1%
1(искажённая)	2%	<b>94%</b>	3%	4%	4%	2%	1%
2	3%	2%	<b>96%</b>	2%	3%	2%	0%
2(искажённая)	2%	3%	<b>95%</b>	3%	2%	2%	1%
3	2%	2%	2%	<b>94%</b>	0%	4%	3%
3(искажённая)	1%	2%	1%	<b>95%</b>	1%	2%	2%
4	1%	4%	1%	1%	<b>94%</b>	2%	2%
4(искажённая)	1%	3%	1%	1%	<b>96%</b>	1%	3%
5	2%	1%	3%	2%	1%	<b>95%</b>	4%
5(искажённая)	1%	1%	2%	1%	1%	<b>95%</b>	3%
6	5%	1%	1%	4%	3%	2%	<b>94%</b>
6(искажённая)	3%	2%	1%	3%	2%	3%	<b>95%</b>

Время вычисления одного поколения при использовании смешанного метода обучения составляет в среднем 1 секунду.

### ЗАКЛЮЧЕНИЕ

В вышеперечисленных данных мы можем заметить, что лучший результат даёт метод обратного распространения ошибки, генетический алгоритм наоборот даёт самый нестабильный результат, так как генетический алгоритм требует подстройку под каждую задачу и по своей сути основан на случайностях; смешанный метод обучения дал максимально приближенный результат к методу обратного распространения ошибки, что говорит нам о том, что если доработать генетический алгоритм, то смешанный метод может оказаться самым эффективным.

Сфера нейронных сетей имеет огромный потенциал в настоящем, и проблема повышения эффективности их работы является одной из основных. Универсального решения этой проблемы на данный момент не существует и под каждую задачу нужно подбирать подходящий метод обучения и организацию самой нейронной сети. Для нахождения всё лучших методов обучения нужно экспериментировать и изучать различные вариации этих методов.

### Список литературы

1. Редько В.Г. Эволюция, нейронные сети, интеллект: Модели и концепции эволюционной кибернетики / В.Г. Редько. – М.: Ленанд, 2019. – 224 с.
2. Жихарев А.Г., Корсунов Н.И., Маматов Р.А., Щербинина Н.В., Пономаренко С.В. О разработке адаптивной образовательной платформы с использованием технологий машинного обучения // Экономика. Информатика. 2022. Т. 49. № 4. С. 810-819.
3. Deeney I.A., Zhikharev A.G., Klyuchnikov D.A., Shurukhina T.N., Gavrilova T.A. Some aspects of AI-technologies in education // Revista San Gregorio. – 2021. – Vol. 44. – P. 186-197.
4. Хайкин С. Нейронные сети: полный курс / С. Хайкин. – М.: Диалектика, 2019. – 1104 с.
5. Вороновский Г.К. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности. – Харьков: Основа, 1997.
6. Каллан Р. Нейронные сети: Краткий справочник / Р. Каллан. – М.: Вильямс И.Д., 2017. – 288 с.
7. Тим Джонс Программирование искусственного интеллекта в приложениях / Пер. с англ. Осипов А.И. – М.: ДМК Пресс, 2011. – 312 с.

## References

1. Redko V.G. Evolution, neural networks, intelligence: Models and concepts of evolutionary cybernetics / V.G. Redko. – М.: Lenand, 2019. – 224 p.
2. Zhikharev A.G., Korsunov N.I., Mamatov R.A., Shcherbinina N.V., Ponomarenko S.V. On the development of an adaptive educational platform using machine learning technologies // Economics. Computer science. 2022. V. 49. No. 4. P. 810-819.
3. Deeney I.A., Zhikharev A.G., Klyuchnikov D.A., Shurukhina T.N., Gavrilova T.A. Some aspects of AI-technologies in education // Revista San Gregorio. – 2021. – Vol. 44. – P. 186-197.
4. Khaikin S. Neural networks: full course / S. Khaikin. – М.: Dialectics, 2019. – 1104 p.
5. Voronovsky G.K. Genetic algorithms, artificial neural networks and problems of virtual reality. – Kharkov: Osnova, 1997.
6. Callan R. Neural networks: A quick guide / R. Callan. – М.: Williams I.D., 2017. – 288 p.
7. Tim Jones Programming artificial intelligence in applications / Per. from English. Osipov A.I. – М.: DMK Press, 2011. – 312 p.

**Черных Владимир Сергеевич**, студент 4 курса направления «Информатика и вычислительная техника», кафедра программного обеспечения вычислительной техники и автоматизированных систем, институт энергетики, информационных технологий и управляющих систем

**Жихарев Александр Геннадиевич**, доктор технических наук, доцент, доцент кафедры программного обеспечения вычислительной техники и автоматизированных систем

**Федосеев Артемий Дмитриевич**, студент 4 курса направления «Информатика и вычислительная техника», кафедра программного обеспечения вычислительной техники и автоматизированных систем, институт энергетики, информационных технологий и управляющих систем

**Мартон Никита Андреевич**, студент 4 курса направления «Информатика и вычислительная техника», кафедра программного обеспечения вычислительной техники и автоматизированных систем, институт энергетики, информационных технологий и управляющих систем

**Chernykh Vladimir Sergeevich**, 4th year student, direction "Computer Science and Computer Engineering", Department of Software for Computer Engineering and Automated Systems, Institute of Energy, Information Technology and Control Systems  
**Zhikharev Alexander Gennadievich**, Doctor of Technical Sciences, Associate Professor, Associate Professor of the Department of Software for Computer Engineering and Automated Systems

**Fedoseev Artemy Dmitrievich**, 4th year student, direction "Computer Science and Computer Engineering", Department of Software for Computer Engineering and Automated Systems, Institute of Energy, Information Technology and Control Systems

**Marton Nikita Andreevich**, 4th year student, direction "Computer Science and Computer Engineering", Department of Software for Computer Engineering and Automated Systems, Institute of Energy, Information Technology and Control Systems