



<http://dx.doi.org/10.35596/1729-7648-2026-24-4-81-88>

УДК 004.032.26

ВЫБОР АРХИТЕКТУРЫ НЕЙРОННОЙ СЕТИ В ЗАДАЧЕ КЛАССИФИКАЦИИ

В. В. МАЦКЕВИЧ

Белорусский государственный университет (Минск, Республика Беларусь)

Аннотация. Нейросетевая обработка данных большой размерности требует значительных вычислительных ресурсов. Неверный выбор архитектуры нейронной сети или преобразования данных может привести к существенным потерям времени и ресурсов на повторное обучение сети. В статье предлагается алгоритм выбора архитектуры сжимающих слоев глубокой доверительной сети и преобразования данных на основе анализа многомерной матрицы несоответствий. Введенный обобщенный коэффициент перекоса описывает равномерность распределения объектов между классами нейронной сети и позволяет численно оценить эффективность архитектуры сжимающих слоев сети и преобразования данных. На примере прикладной задачи показано, что предложенный алгоритм повышает качество ее решения более чем на 10 % путем выбора наиболее эффективной архитектуры нейросети.

Ключевые слова: входные данные, нейронные сети, обучение, архитектура, преобразование данных, многомерная матрица несоответствий, обобщенный коэффициент перекоса.

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Для цитирования. Мацкевич, В. В. Выбор архитектуры нейронной сети в задаче классификации / В. В. Мацкевич // Доклады БГУИР. 2026. Т. 24, № 4. С. 81–88. <http://dx.doi.org/10.35596/1729-7648-2026-24-4-81-88>.

NEURAL NETWORK ARCHITECTURE SELECTION IN CLASSIFICATION PROBLEM

VADIM MATSKEVICH

Belarusian State University (Minsk, Republic of Belarus)

Abstract. Neural network processing of high-dimensional data requires significant computing resources. An incorrect choice of neural network architecture or data transformation can lead to significant losses of time and resources in network retraining. This article proposes an algorithm for selecting the architecture of compressive layers for a deep belief network and data transformation based on the analysis of a multi-label confusion matrix. The introduced generalized skewness coefficient describes the uniformity of the distribution of objects between neural network classes and allows for a numerical evaluation of the effectiveness of the network's compressive layer architecture and data transformation. Using an applied problem as an example, it is demonstrated that the proposed algorithm improves the quality of its solution by more than 10 % by selecting the most effective neural network architecture.

Keywords: input data, neural networks, training, architecture, data transformation, multi-label confusion matrix, generalized skewness coefficient.

Conflict of interests. The author declares that there is no conflict of interests.

For citation. Matskevich V. (2026) Neural Network Architecture Selection in Classification Problem. *Doklady BGUIR*. 24 (4), 81–88. <http://dx.doi.org/10.35596/1729-7648-2026-24-4-81-88> (in Russian).

Введение

Существует множество различных методов преобразования входных данных для нейросетевой обработки: евклидова нормировка, деление на наибольшее значение по модулю, вычитание медианного значения, вычитание среднего значения и т. п. Однако разные преобразования демонстрируют различную эффективность для всевозможных архитектур и решаемых прикладных задач. Более того, для обработки данных большой размерности применяются сжимающие слои глубокой нейронной сети различной архитектуры. Полный перебор всех возможных способов преобразования входных данных с последующим обучением классификатора для проверки эффективности – невыполнимая задача, так как обучение требует огромного объема вычислений.

В статье предлагается алгоритм оценки эффективности на основе многомерной матрицы несоответствий [1]. В отличие от исследований [2], данный алгоритм извлекает больший объем информации о функционировании нейронной сети и не требует ее обучения целиком для проверки эффективности. Кроме того, алгоритм может быть применен к любой архитектуре нейронной сети и решаемой прикладной задаче, что делает его полезным при использовании нейросетевой обработки данных.

Анализ проблемы

Для обработки данных большой размерности существует множество различных глубоких нейронных сетей. Однако все они разделяются на две большие части: сжимающие слои – отвечающие за преобразование входных данных до нужной размерности, очистку от шумов и т. д., и классификатор – многослойный персептрон, который определяет класс входных данных.

Выбор архитектуры сжимающих слоев – трудная задача. Слишком высокая степень сжатия приводит к слишком большим потерям данных и делает последующую классификацию невозможной. Очень низкая степень сжатия порождает входные образы большой размерности, что существенно усложняет задачу классификации, требуя большего объема данных и размера классификатора.

Проверить эффективность сжимающих слоев без обучения классификатора практически невозможно. Ввиду этого, неверно выбранная архитектура сжимающих слоев приводит к необходимости повторного обучения всей глубокой нейронной сети с другой архитектурой, что требует большого количества времени и вычислительных ресурсов.

В настоящее время проводится ряд исследований по архитектурам нейронных сетей, нацеленных на сжатие данных для последующей классификации [3–7]. В [3, 4, 7] предлагается автоматизация выбора архитектуры сжимающих слоев для ускорения разработки нейронной сети. Однако в данных работах выбор архитектуры рассматривается как отдельная сложная оптимизационная задача. Таким образом, проблема выбора сжимающих слоев является актуальной. Правильный выбор сжимающих слоев сети позволяет существенно повысить точность классификации и сократить время разработки сети. Для этого в данной статье введено понятие коэффициента перекоса на основе многомерной матрицы несоответствий, позволяющего оценить эффективность сжимающих слоев без обучения классификатора.

Коэффициент перекоса

Рассмотрим решение задачи классификации с помощью нейронной сети. Для простоты изложения материала без потери общности будем считать, что нейронная сеть задается многослойным персептроном. Пусть в исходной задаче задано n классов входных данных. Классификацию будем производить стандартным образом: какой нейрон выходного слоя обладает наибольшим выходным значением, такому классу и принадлежит обрабатываемый объект.

Пусть задана квадратная матрица $n \times n$, где a_{ij} – количество раз, когда i -й класс выбран j -м нейроном выходного слоя. Элементы матрицы вычисляются по простому алгоритму из трех шагов.

Шаг 1. Элементы матрицы устанавливаются равными нулю.

Шаг 2. Запускается циклическая классификация объектов выборки, на которой вычисляется указанная матрица.

Шаг 2.1. Пусть некоторый r -й объект, принадлежащий классу i , был отнесен нейросетью в класс j . Тогда значение элемента матрицы a_{ij} увеличивается на единицу.

Шаг 3. После обработки всех объектов выборки цикл завершается, а значения элементов матрицы являются результатом вычислений.

Эта матрица также известна как многомерная матрица несоответствий [1] (или multi-label confusion matrix), применяемая для оценки качества конечного решения. Сумма элементов главной диагонали определяет количество правильных ответов сети, сумма всех остальных элементов матрицы – количество неверных классификаций. Матрица также показывает, для каких классов были допущены ошибки, сколько раз и какой нейрон сети был выбран ею для конкретного класса ошибочно.

Воспользуемся данными свойствами матрицы и покажем, что ее можно использовать для численной оценки эффективности нейронной сети. Сумма элементов одного столбца матрицы определяет количество «срабатываний» одного нейрона выходного слоя сети для различных входных данных. С учетом последнего факта построим алгоритм проверки эффективности преобразования входных данных для нейронных сетей. Для этого введем следующее определение.

Определение. Коэффициентом перекоса q многомерной матрицы несоответствий A называется величина, определяющая отношение наибольшей суммы элементов столбца матрицы к сумме всех элементов:

$$q = \frac{\max_{j=1, n} \sum_{i=1}^n a_{ij}}{\sum_{i=1}^n \sum_{j=1}^n a_{ij}}. \quad (1)$$

Величина q для задачи из n классов может варьироваться от $1/n$ до 1. Чем меньше значение q , тем, очевидно, лучше «сбалансированы» входные данные для нейронной сети. Значение $q = 1$ указывает на то, что нейронная сеть полностью «невосприимчива» к входным данным из различных классов. В [2] показано, что данный коэффициент позволяет оценить эффективность архитектуры классификатора. Однако для более точной численной оценки эффективности далее в статье приведен улучшенный вариант коэффициента перекоса.

Обобщенный коэффициент перекоса

Коэффициент перекоса в (1) оценивает соотношение наибольшего класса, созданного нейронной сетью, и размера выборки. Однако с помощью многомерной матрицы несоответствий можно построить более точный инструмент оценки разделения пространства входных данных.

Определение. Обобщенным коэффициентом перекоса u многомерной матрицы несоответствий A называется величина, заданная следующим выражением:

$$u = \sqrt{\sum_{j=1, n} \left(\frac{\sum_{i=1}^n a_{ij}}{\sum_{i=1}^n \sum_{j=1}^n a_{ij}} - \frac{1}{n} \right)^2}. \quad (2)$$

Физический смысл обобщенного коэффициента перекоса состоит в следующем. В классической задаче распознавания для каждого класса количество объектов одинаково. Введенный коэффициент вычисляет евклидово расстояние от «идеального» распределения объектов между классами. То есть сначала вычисляется квадрат отклонения от «нормы» по каждому из классов, формируя тем самым многомерный вектор из n координат, а затем – длина данного вектора.

Следует отметить, что обобщенный коэффициент u отличается от обычного q . Во-первых, введена нормировка, которая вычитает равномерное распределение объектов между классами (второе слагаемое в скобках), что является стандартом для задач классификации. Во-вторых, u извлекает гораздо больше информации из распределения классов, учитывая распределение по всем классам, а не по наибольшему.

Покажем, что коэффициент перекоса u является обобщением использованного в предыдущих исследованиях коэффициента q . Для этого сформулируем следующее утверждение.

Утверждение. В задаче классификации из n классов изменение значения коэффициента перекоса q гарантирует изменение обобщенного коэффициента перекоса u , при этом обратное не верно при $n > 2$.

Доказательство. Начнем доказательство со второй части утверждения. Пусть вычислены отношения сумм столбцов к общей сумме элементов многомерной матрицы несоответствий $(b_1, b_2, \dots, b_n)$, а некоторый столбец i определяет максимальное значение – коэффициент перекоса. Выполнив преобразования, заменим все соотношения столбцов, кроме i -го, следующим выражением:

$$b'_j = \frac{b_j}{\sum_{k=1, n, k \neq i} b_k}.$$

Вычислим обобщенный коэффициент перекоса для старых b_j и преобразованных b'_j значений. Если все значения после преобразования не изменились (т. е. были изначально равны между собой), то и величина u не изменилась. Однако, если хотя бы одно из значений до преобразования было не равно остальным, то преобразование производит «сглаживание», что, очевидно, уменьшает значение обобщенного коэффициента перекоса. При этом величина q остается неизменной, поскольку максимальный столбец после «сглаживания» не изменился. Что и требовалось доказать.

Теперь докажем основную часть утверждения, а именно, что изменение значения коэффициента перекоса всегда приводит к изменению величины обобщенного коэффициента перекоса. Для $n = 2$ утверждение очевидно, так как увеличение значения соотношения в одном столбце автоматически понижает значение во втором. Если увеличено значение для столбца с показателем больше, чем 0,5, то увеличиваются значения обоих коэффициентов перекоса, поскольку у другого столбца с показателем менее 0,5 значение становится еще меньше. Верно и в обратную сторону. Если величина коэффициента перекоса q уменьшается, то уменьшается и значение u , так как уменьшение показателя в столбце со значением более 0,5 увеличивает значения для столбца, у которого соотношение было меньше 0,5, что понижает отклонение от равномерного распределения.

Для $n > 2$ рассмотрим случай уменьшения коэффициента перекоса. Пусть заданы значения соотношений для столбцов $b_1, b_2, \dots, b_n$, и некоторый i -й столбец определяет величину коэффициента перекоса. Выполнили некоторое преобразование, в результате которого значение коэффициента перекоса уменьшилось. Пусть после преобразования j столбец определяет значение коэффициента перекоса. При этом, возможно, $i = j$. Среди допустимых преобразований рассмотрим такое, которое максимизирует значение обобщенного коэффициента перекоса. Если для него показать, что оно уменьшает величину u , то и для всех остальных допустимых преобразований значение коэффициента уменьшится.

Рассмотрим «жадную» стратегию увеличения обобщенного коэффициента перекоса. Уменьшение его величины означает, что максимальное значение для столбцов было уменьшено. При этом по условию задачи классификации данное уменьшение должно быть добавлено к остальным столбцам по двум правилам:

1) максимальное значение для столбцов должно быть меньше после прибавления по предположению;

2) для преобразованных значений b'_j , как и для исходных по определению, верно следующее соотношение:

$$\sum_{i=1, n} b_i = \sum_{i=1, n} b'_i = 1.$$

Пусть преобразование уменьшило значение коэффициента перекоса на величину δ . Тогда для максимизации значения обобщенного коэффициента перекоса необходимо добавить данную величину для столбца b_j с наибольшим для остальных столбцов значением. Если $(b_j + \delta) \leq (b_i - \delta)$, то все остальные столбцы не изменяются, а величина обобщенного коэффициента уменьшается, поскольку для отдельных его слагаемых (столбцов i, j) сумма значений уменьшилась (очевидно, что $\left(b_i - \frac{1}{n}\right)^2 + \left(b_j - \frac{1}{n}\right)^2 > \left(b_i - \delta - \frac{1}{n}\right)^2 + \left(b_j + \delta - \frac{1}{n}\right)^2$). Если же $(b_j + \delta) > (b_i - \delta)$, то для обеспечения уменьшения коэффициента перекоса необходимо вычислить $\delta' = (b_j + \delta) - (b_i - \delta)$, а затем добавить к столбцу b_j величину $\delta - \delta'$ и повторить «жадную» стратегию снова. При добавлении

к столбцу b_j величины $\delta - \delta'$ и отнимании от b_i этого значения уменьшается значение обобщенного коэффициента, и, следовательно, оно строго убывает. Что и требовалось показать.

Случай с увеличением значения коэффициента перекоса симметричен и не требует доказательства.

Таким образом, показано, что обобщенный коэффициент перекоса дает более точную оценку распределения объектов между классами по сравнению с простым коэффициентом. При этом объемом дополнительных вычислений, по сравнению с вычислением обычного коэффициента, можно пренебречь.

Алгоритм оценки эффективности архитектуры

При исследовании свойств коэффициента перекоса установлено, что он позволяет оценить эффективность архитектуры классификатора в зависимости от решаемой задачи. Эксперименты показали, что чем меньше значение коэффициента для классификатора, тем выше качество конечного решения можно получить в процессе обучения [2]. Воспользуемся данным фактом для проектирования алгоритма оценки эффективности сжимающих слоев. Для этого рассмотрим следующий набор этапов.

Этап 1. Обучение слоев. На данном шаге обучаются все сжимающие слои, из которых будет осуществляться выбор архитектуры.

Этап 2. Проектирование классификатора. Определяется архитектура многослойного персептрона: количество и размеры слоев, функции активации и алгоритм инициализации параметров. Размер входного слоя варьируется в зависимости от размерности входных данных после обработки сжимающими слоями.

Этап 3. Оценка эффективности слоев. Вычисляется усредненное значение обобщенного коэффициента перекоса для всех сравниваемых сжимающих слоев. Для этого данные пропускаются через все сравниваемые слои, а затем вычисляется коэффициент для многослойного персептрона после каждого сжатия.

Этап 4. Выбор оптимального сжатия. Выбираются сжимающие слои, обеспечивающие наименьшее среднее значение коэффициента.

Замечание. Обобщенный коэффициент перекоса обладает большой дисперсией, поэтому рекомендуется его вычислять по усреднению 1000 замеров.

Хотя предложенный алгоритм и требует обучения всех сравниваемых сжимающих слоев, он позволяет сократить время разработки сети, отбрасывая обучение персептрона для каждой архитектуры сети, заменяя ее на определение коэффициента перекоса, требующее гораздо меньшего объема вычислений. Алгоритм допускает модификацию для оценки эффективности преобразования данных без использования сжимающих слоев. Для этого достаточно убрать этап 1, а этап 3 заменить на оцениваемое преобразование данных.

Таким образом, предлагаемый алгоритм позволяет численно оценить качество преобразования входных данных, в том числе сжимающих слоев сети, без обучения классификатора, что существенно ускоряет перебор возможных архитектур и разработку сети. Для обоснования корректности такого алгоритма рассмотрим пространство входных данных. В задачах классификации предполагается, что содержится равное количество объектов всех классов, и мощности занимаемых ими подпространств равны.

Расчет обобщенного коэффициента перекоса показывает, насколько распределение объектов между классами, построенное нейронной сетью, «далеко» от предполагаемого равномерного. Если рассмотреть усредненное значение u , можно отметить, что он оценивает все возможные разделяющие поверхности, которые способна построить нейронная сеть рассматриваемой архитектуры. При 100%-ной классификации обобщенный коэффициент перекоса будет равен нулю, поскольку нейронная сеть точно воспроизводит равномерное распределение объектов между классами. Чем больше разделяющих поверхностей, близких к равномерному распределению объектов между классами, может построить нейронная сеть, тем меньше значение усредненного u и тем выше вероятность того, что обученная сеть будет иметь наибольшую возможную в условиях задачи обобщающую способность.

Таким образом, на основе проведенных исследований можно сделать вывод, что представленный алгоритм является эвристическим, поскольку доказана обратная корреляция обобщающей способности сети и значения обобщенного коэффициента перекоса.

Проведение экспериментов

В рамках экспериментов рассматривали прикладную задачу детектирования непогодных изменений поверхности Земли по результатам анализа спутниковых снимков [8]. Спутниковые снимки разбивали на блоки 20×20 пикселей и решали задачи независимо для каждого блока. С учетом масштаба снимков блоки указанного размера вмещали наименьший возможный объект исследования. Для простоты решали задачу с использованием простого многослойного персептрона. Поскольку по условию задачи непогодные изменения определялись на основе пары снимков одной и той же местности в разные моменты времени, на вход нейронной сети поступали два цветных изображения разрешением 20×20 пикселей. Следовательно, входной слой содержал 2400 нейронов, а выходной – всегда два с функцией активации softmax (один нейрон указывал на наличие непогодных изменений, второй – на их отсутствие). В скрытом слое располагалось 64 нейрона с биполярно-сигмоидной функцией активации.

Архитектура нейронной сети фиксировалась на время проведения экспериментов, так как для разных архитектур могли «подойти» различные преобразования входных данных. Веса нейронной сети инициализировались как реализации равномерно распределенных на отрезке $[-0,05, 0,05]$ случайных величин, смещения нейронов задавались нулем. Для обучения применялся метод отжига, приведенный в [9]. Для обучения использовался компьютер на ОС Arch Linux с видеокартой nvidia rtx 3070, процессором ryzen 7 7700 и 32 GB DDR5 оперативной памяти.

Вначале проводили сравнение алгоритмов преобразования входных данных. Обобщенный коэффициент перекося вычислялся путем усреднения 1000 повторов. Для сравнения качества решения была выбрана относительная шкала. За единицу принимали наилучшее решение. Дополнительно добавляли столбец с вычислением коэффициента перекося (табл. 1).

Таблица 1. Результаты преобразования входных данных
Table 1. Input data transformation results

Алгоритм преобразования	Коэффициент перекося	Обобщенный коэффициент перекося	Качество решения
Евклидова нормировка, вычитание медианы	0,5997	0,1464	0,7070
Евклидова нормировка, вычитание среднего	0,5938	0,1396	0,7152
Евклидова нормировка	0,9039	0,5813	0,7058
Вычитание медианы и деление на константу	0,5727	0,1053	1,0000
Вычитание среднего и деление на константу	0,5767	0,1082	0,998

По результатам экспериментов можно отметить, что чем ниже среднее значение обобщенного коэффициента перекося, тем выше качество конечного решения. Кроме того, усредненное значение коэффициента перекося заметно менее чувствительное, чем у обобщенного коэффициента. Количество повторов замеров коэффициента можно сократить на порядок, если обучающая выборка большого размера.

В процессе исследований проверяли алгоритм выбора сжимающих слоев на основе значения обобщенного коэффициента перекося. Для чистоты проведения экспериментов, исключая дефекты входных данных, использовали выборку CIFAR-10. В качестве сжимающего слоя применялись ансамбли из автоэнкодеров и ограниченных машин Больцмана типа Гаусс – Бернулли. Каждая из архитектур обрабатывала независимо фрагменты изображений размерами 4×4 пикселей с целью сжатия данных и подавления шумов. Первая архитектура представляла собой ансамбль из автоэнкодеров с 48 нейронами во входном и 24 нейронами в скрытом слоях, обеспечивающих двукратное уменьшение размерности входных данных. Этот подход к сжатию применяется в MLP-миксерах [10, 11] и в сверточных сетях в виде слоев свертки [12, 13]. Вторая архитектура была представлена в виде ансамбля из ограниченных машин Больцмана типа Гаусс – Бернулли с 48 нейронами во входном и 192 нейронами в скрытом слоях, обеспечивая четырехкратное сжатие. Третья архитектура – ансамбль из ограниченных машин Больцмана типа Гаусс – Бернулли с 48 нейронами во входном и 96 нейронами в скрытом слоях, обеспечивая восьмикратное сжатие.

После сжимающих слоев располагался многослойный персептрон с одним скрытым слоем из 64 нейронов с функцией активации GELU [14], наиболее эффективной для глубоких нейронных сетей, и 10 нейронами в выходном слое с функцией активации softmax. Для сравнения качества решения была выбрана относительная шкала. За единицу принимали наилучшее решение (табл. 2).

Таблица 2. Результаты выбора архитектуры сжимающих слоев
Table 2. Results of choosing the architecture of compression layers

Архитектура	Коэффициент перекося	Обобщенный коэффициент перекося	Качество решения
Автоэнкодер, двукратное сжатие	0,1988	0,1521	1,0000
Ограниченная машина Больцмана: четырёхкратное сжатие	0,4160	0,4030	0,8023
восьмикратное сжатие	0,3619	0,3463	0,8586

По результатам экспериментов установлено, что чем меньше значение обобщенного коэффициента перекося, тем выше качество конечного решения. Правильный выбор архитектуры сжимающих слоев позволил повысить качество конечного решения на 11,2 %. Обучение классификатора потребовало от 1 до 2 ч в зависимости от архитектуры сжимающих слоев. В то же время вычисление коэффициента перекося (в том числе и обобщенного) требует простого прохождения данных через нейронную сеть. Поскольку сжимающие слои фиксировались при повторных вычислениях коэффициента перекося, результат сжатия можно закешировать и повторять обработку только для многослойных перцептронов, что еще больше уменьшает объем вычислений и сокращает время выбора архитектуры. Кроме того, обобщенный коэффициент перекося позволяет не только выбрать эффективную архитектуру классификатора, но и сжимающих слоев и алгоритма преобразования входных данных, что может быть полезно при проектировании глубоких нейронных сетей с большим количеством параметров.

Заключение

1. Рассмотрена проблема выбора архитектуры нейронной сети при решении задачи классификации. Предложен алгоритм выбора преобразования входных данных и сжимающих слоев сети на основе усредненного значения обобщенного коэффициента перекося.

2. Показано, что обобщенный коэффициент перекося не требует большого объема вычислений для любой архитектуры и решаемой прикладной задачи.

3. Предложенный алгоритм позволяет повысить качество решения более чем на 10 % за счет выбора наиболее эффективной архитектуры.

Список литературы

- Heydarian, M. MLCM: Multi-Label Confusion Matrix / M. Heydarian, T. E. Doyle, R. Samavi // IEEE Access. 2022. Vol. 10. P. 19083–19095. <http://dx.doi.org/10.1109/ACCESS.2022.3151048>.
- Мацкевич, В. В. Адаптация нейросетевого классификатора к неустойчивым входным данным / В. В. Мацкевич, Ц. Го // Доклады БГУИР. 2025. Т. 23, № 5. С. 99–104. <http://dx.doi.org/10.35596/1729-7648-2025-23-5-99-104>.
- AutoPolCNN: A Neural Architecture Search Method of Convolutional Neural Network for Polars Image Classification / G. Liu [et al.] // Knowledge-Based Systems. 2025. Vol. 312. <https://doi.org/10.1016/j.knosys.2025.113122>.
- Automatic Neural Network Construction Based on Neural Tangent Kernel for IRS-Aided Beamforming / H. Shi [et al.] // IEEE Transactions on Wireless Communications. 2025. Vol. 25. P. 1155–1168. <https://doi.org/10.1109/TWC.2025.3589367>.
- Synthesis of Convolutional Neural Network Architectures for Biomedical Image Classification / O. Berezhsky [et al.] // Biomedical Signal Processing and Control. 2024. Vol. 95, Part B. <https://doi.org/10.1016/j.bspc.2024.106325>.
- Akay, B. A Comprehensive Survey on Optimizing Deep Learning Models by Metaheuristics / B. Akay, D. Karaboga, R. Akay // Artificial Intelligence Review. 2022. Vol. 55, Iss. 2. <https://doi.org/10.1007/s10462-021-09992-0>.
- Xi, Sh. Dynamic Gaussian Deep Belief Network Design and Stock Market Application / Sh. Xi, X. Xu // Intelligent Data Analysis. 2023. Vol. 27, Iss. 2. P. 519–534. <http://dx.doi.org/10.3233/ida-216340>.
- Zhou, X. Landscape's Non-Natural Changes Detection System by Satellites Images Based on Local Areas / X. Zhou [et al.] // Pattern Recognition and Image Analysis. Advances in Mathematical Theory and Applications. 2024. Vol. 34, Iss. 2. P. 365–378. <https://doi.org/10.1134/S1054661824700159>.
- Krasnoprosin, V. V. Random Search in Neural Networks Training / V. V. Krasnoprosin, V. V. Matskevich // Pattern Recognition and Image Analysis. 2024. Vol. 34, Iss. 2. P. 309–316. <https://doi.org/10.1134/S105466182470010X>.
- Tang, H. A Pure MLP-Mixer-Based GAN Framework for Guided Image Translation / H. Tang, B. Ren, N. Sebe // Pattern Recognition. 2025. Vol. 157. <https://doi.org/10.1016/j.patcog.2024.110894>.

11. Multi-Scale MLP-Mixer for Image Classification. / H. Zhang [et al.] // Knowledge-Based Systems. 2022. Vol. 258. <https://doi.org/10.1016/j.knosys.2022.109792>.
12. Raj, R. An Extensive Study of Convolutional Neural Networks: Applications in Computer Vision for Improved Robotics Perceptions / R. Raj, A. Kos // Sensors. 2025. Vol. 25, Iss. 4. <https://doi.org/10.3390/s25041033>.
13. Abdellatef, E. Detection of Human Activities Using Multi-Layer Convolutional Neural Network / E. Abdellatef, R. M. Al-Makhlasy, W. A. Shalaby // Scientific Reports. 2025. Vol. 15, Iss. 1. <https://doi.org/10.1038/s41598-025-90307-6>.
14. Lee, M. GELU Activation Function in Deep Learning: A Comprehensive Mathematical Analysis and Performance / M. Lee // arXiv. 2023. <https://doi.org/10.48550/arXiv.2305.12073>.

Поступила 02.06.2026

Принята в печать 29.06.2026

References

1. Heydarian M., Doyle T. E., Samavi R. (2022) MLCM: Multi-Label Confusion Matrix. *IEEE Access*. 10, 19083–19095. <http://dx.doi.org/10.1109/ACCESS.2022.3151048>.
2. Matskevich V. V., Guo J. (2025) Neural Network Classifier Adaptation to Unstable Input Data. *Doklady BGUIR*. 23 (5), 99–104. <http://dx.doi.org/10.35596/1729-7648-2025-23-5-99-104> (in Russian).
3. Liu G., Li Y., Chen Y., Shang R., Jiao L. (2025) AutoPolCNN: A Neural Architecture Search Method of Convolutional Neural Network for PolSAR Image Classification. *Knowledge-Based Systems*. 312. <https://doi.org/10.1016/j.knosys.2025.113122>.
4. Shi H., Ji T., Wang Z., Jin S., Huang Y. (2025) Automatic Neural Network Construction Based on Neural Tangent Kernel for IRS-Aided Beamforming. *IEEE Transactions on Wireless Communications*. 25, 1155–1168. <https://doi.org/10.1109/TWC.2025.3589367>.
5. Berezsky O., Liashchynskyi P., Pitsun O., Izonin I. (2024) Synthesis of Convolutional Neural Network Architectures for Biomedical Image Classification. *Biomedical Signal Processing and Control*. 95 (B). <https://doi.org/10.1016/j.bspc.2024.106325>.
6. Akay B., Karaboga D., Akay R. (2022) A Comprehensive Survey on Optimizing Deep Learning Models by Metaheuristics. *Artificial Intelligence Review*. 55 (2). <https://doi.org/10.1007/s10462-021-09992-0>.
7. Xi Sh., Xu X. (2023) Dynamic Gaussian Deep Belief Network Design and Stock Market Application. *Intelligent Data Analysis*. 27 (2), 519–534. <http://dx.doi.org/10.3233/ida-216340>.
8. Zhou X., Bu Q., Matskevich V., Nedzved A. (2024) Landscape's Non-Natural Changes Detection System by Satellites Images Based on Local Areas. *Pattern Recognition and Image Analysis*. 34 (2), 365–378. <http://dx.doi.org/10.1134/S1054661824700159>.
9. Krasnoprosin V. V., Matskevich V. V. (2024) Random Search in Neural Networks Training. *Pattern Recognition and Image Analysis*. 34 (2), 309–316. <https://doi.org/10.1134/S105466182470010X>.
10. Tang H., Ren B., Sebe N. (2025) A Pure MLP-Mixer-Based GAN Framework for Guided Image Translation. *Pattern Recognition*. 157. <https://doi.org/10.1016/j.patcog.2024.110894>.
11. Zhang H., Dong Z., Li B., He S. (2022) Multi-Scale MLP-Mixer for Image Classification. *Knowledge-Based Systems*. 258. <https://doi.org/10.1016/j.knosys.2022.109792>.
12. Raj R., Kos A. (2025) An Extensive Study of Convolutional Neural Networks: Applications in Computer Vision for Improved Robotics Perceptions. *Sensors*. 25 (4). <https://doi.org/10.3390/s25041033>.
13. Abdellatef E., Al-Makhlasy R. M., Shalaby W. A. (2025) Detection of Human Activities Using Multi-Layer Convolutional Neural Network. *Scientific Reports*. 15 (1). <https://doi.org/10.1038/s41598-025-90307-6>.
14. Lee M. (2023) GELU Activation Function in Deep Learning: A Comprehensive Mathematical Analysis and Performance. *arXiv*. <https://doi.org/10.48550/arXiv.2305.12073>.

Received: 2 June 2026

Accepted: 29 June 2026

Сведения об авторе

Мацкевич В. В., канд. техн. наук, доц. каф. информационных систем управления, Белорусский государственный университет

Адрес для корреспонденции

220030, Республика Беларусь,
Минск, просп. Независимости, 4
Белорусский государственный университет
Тел.: +375 29 125-49-07
E-mail: matskevich1997@gmail.com
Мацкевич Вадим Владимирович

Information about the author

Matskevich V., Cand. Sci. (Tech.), Associate Professor of the Department of Information Management Systems, Belarusian State University

Address for correspondence

220030, Republic of Belarus,
Minsk, Nezavisimosti Ave., 4
Belarusian State University
Тел.: +375 29 125-49-07
E-mail: matskevich1997@gmail.com
Matskevich Vadim