

УДК 004.89  
DOI: 10.15827/0236-235X.132.591-598

Дата подачи статьи: 29.06.20  
2020. Т. 33. № 4. С. 591–598

## **Реализация программных средств для классификации данных на основе аппарата сверточных нейронных сетей и прецедентного подхода**

*П.Р. Варшавский*<sup>1</sup>, к.т.н., доцент, [VarshavskyPR@mpei.ru](mailto:VarshavskyPR@mpei.ru)  
*А.В. Кожевников*<sup>1</sup>, ассистент, [antoko@yandex.ru](mailto:antoko@yandex.ru)

<sup>1</sup> *Национальный исследовательский университет «МЭИ»,  
кафедра прикладной математики и искусственного интеллекта,  
г. Москва, 111250, Россия*

Статья посвящена вопросам реализации программных средств для классификации данных, использующих рассуждения на основе прецедентов (Case-Based Reasoning, CBR) и технологию сверточных нейронных сетей (Convolutional Neural Network, CNN). В настоящее время CBR-методы широко используются для поиска решения различных задач на основе накопленного опыта, а CNN успешно применяются при решении задач классификации за счет выделения отдельных элементов и формирования высокоуровневых признаков с использованием ядер свертки.

Одним из необходимых условий успешности решения задачи классификации данных является наличие корректной обучающей выборки. К сожалению, это условие не всегда может быть выполнено (например, в силу сложности рассматриваемых объектов и недостатка исходной информации). Благодаря способности накапливать, использовать и адаптировать имеющийся опыт CBR-методы могут применяться для формирования обучающей выборки, которая в дальнейшем может использоваться другими методами для решения задачи классификации данных. Таким образом, интеграция CBR-методов и CNN позволяет повысить эффективность решения задачи классификации данных.

Помимо этого, CBR-методы могут применяться в областях с непредсказуемым поведением и обучаться в процессе функционирования, например, в процессе обучения нейронных сетей. В данной статье предлагается CBR-метод для обучения CNN, обеспечивающий контроль за обучением нейронной сети, а также прецедентное представление итераций при обучении. Подбор шага обучения CNN на основе прецедентов способствует повышению быстродействия алгоритма обучения сети.

На основе предложенных методов в среде MS Visual Studio на языке C# реализован нейросетевой блок с использованием CNN, расширяющий возможности прецедентной системы (CBR-модуля) для решения задачи классификации данных. Для оценки эффективности предлагаемых в работе решений выполнены вычислительные эксперименты на реальных наборах данных.

**Ключевые слова:** *прецедентный подход, сверточная нейронная сеть, обучение нейронной сети, анализ данных, классификация.*

Одной из основных задач *интеллектуального анализа данных* (ИАД) является задача их классификации. Она позволяет определить, к какому классу относятся те или иные данные при условии, что множество классов, к одному из которых впоследствии можно отнести исследуемый объект, заранее обусловлено. В классическом варианте среди всего множества исследуемых объектов существуют объекты, для которых известно, к какому классу они относятся. Такие объекты называются обучающими примерами, а подмножество обучающих примеров – обучающей выборкой, и задачу именно от ее характеристик зависит качество решения задачи классификации.

Среди основных характеристик обучающей выборки необходимо выделить достаточность,

разнообразие и равномерность. Достаточность обучающей выборки подразумевает, что число обучающих примеров является достаточным для обучения, разнообразие – что среди обучающих примеров имеется большое число разнообразных комбинаций вход-выход, а равномерность – что обучающие примеры различных классов представлены примерно в одинаковых пропорциях.

Большинство подходов ИАД требуют одновременного наличия этих трех характеристик у обучающей выборки, однако далеко не всегда такая обучающая выборка существует. Подготовка обучающей выборки является сложной задачей, с которой порой не может справиться даже эксперт в этой области. Эксперт может иметь четкое понимание разделения объектов

на классы, однако зачастую в условиях пограничных состояний не всегда способен однозначно определить принадлежность объекта к конкретному классу. Для формирования обучающей выборки можно применить прецедентный подход Case-Based Reasoning (CBR) [1, 2], ориентированный на использование и адаптацию накопленного опыта для получения новых правдоподобных суждений.

В данной работе предлагается на основе предложенной экспертом обучающей выборки малого размера, используя прецеденты (CBR-подхода), формировать выборку с одновременным наличием трех вышеуказанных характеристик, которая далее используется для обучения сети CNN (Convolutional Neural Network). Обучение CNN является ресурсоемкой задачей, поэтому для повышения его эффективности предлагается применять CBR-подход. Представление итераций при обучении как прецедентов и настройка на их основе алгоритма обучения CNN (подбор соответствующего шага обучения) способствуют повышению быстродействия алгоритма обучения.

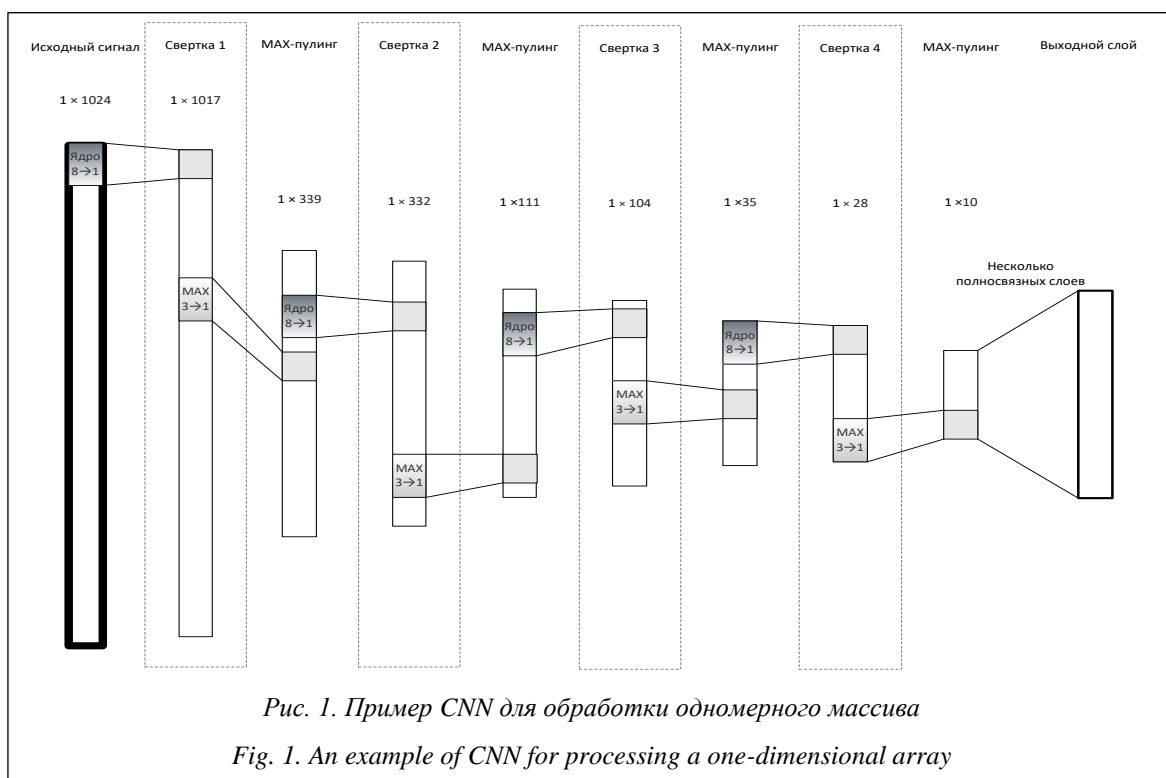
### Сети CNN и их обучение

В 1988 году Ян Лекун предложил специальную архитектуру искусственных нейронных

сетей, нацеленную на эффективное распознавание образов, называемую сверточной [3]. Появление CNN было мотивировано биологическими визуальными системами. В архитектуру CNN закладывается знание из компьютерного зрения: каждый пиксель изображения сильнее связан с соседними пикселями по сравнению с удаленными, а объект на изображении может находиться в любом месте. Главное отличие CNN от других типов нейронных сетей состоит в организации слоев. Выделяют слой свертки и слой пулинга, или субдискретизации [4].

Основным блоком CNN считается слой свертки с небольшим количеством входных нейронов, ядро которого обрабатывает предыдущий слой по фрагментам. После слоя свертки результат обычно передается в слой пулинга, представляющий собой нелинейное уплотнение карты признаков. Пулинг интерпретируется следующим образом: если на предыдущей операции свертки уже были выявлены некоторые признаки, то для дальнейшей обработки настолько подробная информация уже не нужна, и она уплотняется до менее подробной (одним из следствий является фильтрация ненужных деталей).

Как правило, CNN организуют следующим образом (рис. 1): чередуются слои свертки и пулинга, а выходной слой является полносвяз-



ным. К нему на вход подаются признаки, вырабатываемые сложной сетью, малопонятные и трудные для интерпретации. Работа с такими признаками позволила CNN добиться высоких результатов в задачах классификации и кластеризации данных [5, 6].

Основным преимуществом CNN по сравнению с другими типами нейронных сетей является гораздо меньшее количество настраиваемых весовых коэффициентов, так как одно ядро весов используется целиком для всего массива значений, вместо того чтобы для каждой точки входного набора параметров применять свои весовые коэффициенты.

Процесс функционирования нейронной сети зависит от величин синаптических связей, поэтому при решении конкретной задачи важными этапами являются выбор структуры (архитектуры, топологии) сети и определение оптимальных значений всех весовых коэффициентов для синаптических связей. Для обучения нейронных сетей чаще всего применяется алгоритм обратного распространения ошибки [7]. В его основе лежит методика, позволяющая быстро вычислять вектор частных производных (градиент) сложной функции многих переменных, если структура этой функции известна. В качестве такой функции выступает функция ошибки нейронной сети, определяемая архитектурой этой сети.

В случае CNN имеются слои пулинга и свертки, для которых подсчет ошибки выполняется по определенным алгоритмам [7]. Для слоя пулинга подсчет ошибки зависит от функции пулинга: если считается как среднее арифметическое, то ошибка равномерно распределяется по всем нейронам предыдущего слоя, если максимум или минимум, то ошибка приравнивается тому нейрону предыдущего слоя, с которого был получен результат, для остальных нейронов ошибка считается нулевой.

Для подсчета ошибки слоя свертки используется обратная свертка: если в случае прямой свертки количество выходов было меньше количества входов слоя, то в случае обратной свертки необходимо получить ошибки большего количества входов слоя из ошибок меньшего количества выходов. Для этого используется «перевернутое» ядро свертки, полученное с помощью поворота ядра свертки на  $180^\circ$ . К набору значений ошибок выходов слоя дописываются нули «по краям», и выполняется прямая свертка. Таким образом, формируются ошибки элементов ядра свертки [7].

После получения ошибки всех нейронов сети выполняется изменение весов синаптических связей: из текущего значения веса вычитается ошибка, умноженная на некоторый конечный шаг, который называют шагом обучения нейронной сети.

Если в процессе обучения CNN наступает момент, когда ошибка сети попадает в рамки допустимых значений, значит, наблюдается сходимость алгоритма обучения. Если размер шага фиксированный и очень мал, то сходимость нейронной сети слишком медленная, если же он фиксированный и слишком велик, то может возникнуть паралич или постоянная неустойчивость обучения сети.

### Повышение эффективности обучения CNN с помощью прецедентов

С целью повышения эффективности обучения CNN предлагается использовать прецедентный подход [8] для контроля за скоростью обучения. В прецеденте хранится следующая информация:  $CASE = (Situation, Solution, Result)$ , где *Situation* – ситуация, к которой применяется данный прецедент; *Solution* – решение (например, диагноз и рекомендации пользователю); *Result* – результат применения решения, который может включать список выполненных действий, дополнительные комментарии и ссылки на другие прецеденты, а при необходимости обоснование выбора данного решения и возможные альтернативы [9].

В качестве  $Situation_i$  в прецеденте хранятся набор весовых коэффициентов (весов) CNN до последнего шага  $W_{i-1}$ , величина шага  $h_i$ , ошибка до предыдущего шага  $E_{i-1}$ , ошибка после предыдущего шага  $E_i$  и указатель на предшествующий прецедент  $CASE_{i-1}$ , по которому была получена такая ошибка. В  $Solution_i$  содержится рекомендуемое изменение величины шага  $\Delta hr_i$ , а в  $Result_i$  сохраняется ошибка  $Er_i$ , посчитанная после применения данного прецедента, и  $R_i$  – оценка эффективности применения данного прецедента. Таким образом, каждый прецедент хранит в себе информацию о том, как было получено текущее состояние и как оценить качество решения (рекомендаций).

Наличие указателя на предыдущий прецедент  $CASE_{i-1}$  позволяет хранить предысторию появления данного прецедента в виде цепочки прецедентов, элементы которой можно представить в виде некоторого конечного автомата переходов состояний. Существенным отличием от такого конечного автомата является то,

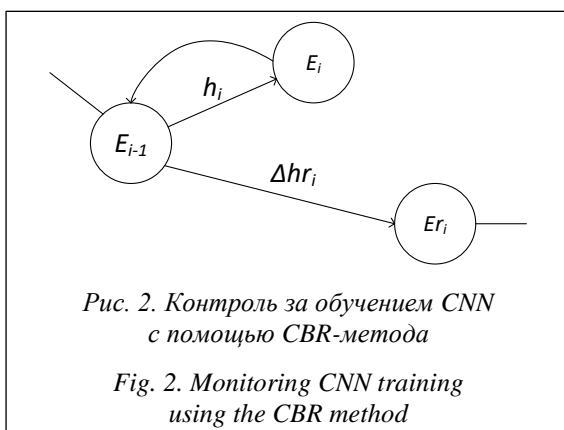
что переход из одного состояния в другое может как повторяться, так и быть адаптированным исходя из ранее полученного опыта: изменения величин шага  $\Delta hr_i$  не просто копируются из схожих прецедентов, а вычисляются в зависимости от изменений величин шага и схожести прецедентов.

Для прецедента с номером  $i$  формируется множество  $S_i$ , составленное из номеров ранее сохраненных прецедентов и схожих с  $i$ -м прецедентом, а изменение шага вычисляется по формуле

$$\Delta hr_i = \frac{1}{|S_i|} \sum_{j=1}^{|S_i|} Sim(CASE_i, CASE_j) * \Delta hr_j,$$

где  $|S_i|$  – количество элементов (номеров схожих прецедентов) в множестве  $S_i$ ;  $Sim(CASE_i, CASE_j)$  – значение функции, определяющей сходство двух прецедентов  $CASE_i$  и  $CASE_j$  ( $i \neq j$ ) (например, с помощью евклидовой метрики);  $\Delta hr_j$  – выполненное изменение шага в прецеденте  $CASE_j$ . В случае отсутствия схожих прецедентов шаг не изменяется.

Предложенный метод способен контролировать процесс обучения CNN и в случае необходимости выполнять его настройку: ускорить или замедлить обучение, а также вернуть веса CNN в предыдущее состояние при увеличении ошибки обучения (рис. 2). Наличие такого механизма позволяет исключить некоторые неудачные примеры (прецеденты) из обучающей выборки, которые ухудшают работу нейронной сети и увеличивают ошибки при обучении.



Однако при таком подходе существует вероятность того, что некоторые прецеденты будут пропущены. Для предотвращения указанной ситуации предлагается накапливать информацию по каждому примеру из обучающей выборки, когда пример был пропущен в процессе обучения. После достижения заданного порогового значения пропускаемый пример

считается критическим. Когда CNN обучилась, все критические примеры проверяются на наличие ошибок классификации. Если в результате требуемое качество классификации не достигается, процесс обучения CNN повторяется.

### Реализация нейросетевого блока на основе CNN для CBR-модуля

В работе выполнена программная реализация на языке C# (.NET 4.5) [10] под операционную систему MS Windows нейросетевого блока на основе CNN для расширения CBR-модуля, ориентированного на решение задач классификации данных. Работа реализованных программных средств рассмотрена на примере решения задачи классификации сигналов (см. <http://www.swsys.ru/uploaded/image/2020-4/2020-4-dop/1.jpg>), полученных в результате акустико-эмиссионного мониторинга сложных технических объектов [1, 2].

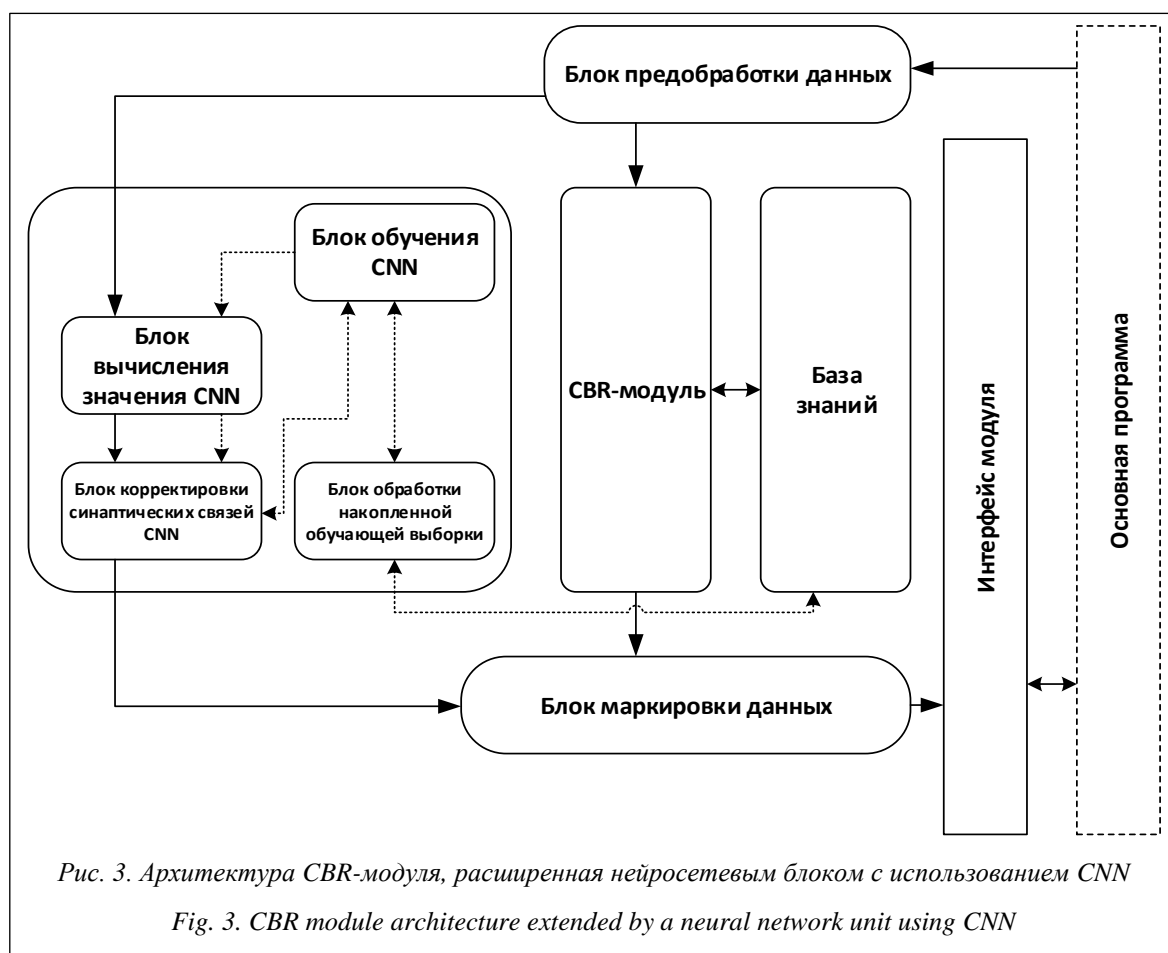
Для реализации функций CNN и CBR при обучении CNN были реализованы следующие программные средства (рис. 3): блок обработки накопленной обучающей выборки, блок вычисления результата с помощью CNN, блок корректировки синаптических связей и блок обучения CNN на основе прецедентов. Наиболее важным компонентом является блок обучения CNN, так как он выполняет функции контроля и управления процессом обучения.

Изначально прецедентный модуль накапливает в базе знаний (базе прецедентов) информацию по классификации объектов, затем дополнительно может активироваться режим обучения CNN, в ходе которого возможно снижение производительности системы. Во время обучения сети CBR-модуль продолжает накапливать прецеденты, однако эти прецеденты уже не участвуют в процессе обучения, но используются в качестве тестовой (проверочной) выборки.

После достижения заданного качества классификации (точности обучения CNN) CBR-модуль прекращает накапливать прецеденты и переключается на классификацию данных с помощью обученной на накопленных прецедентах CNN для уменьшения числа операций на обработку одного объекта.

### Анализ результатов обучения CNN на основе прецедентов

Для оценки применимости предложенного метода обучения CNN на 30 файлах с данными



акустико-эмиссионного мониторинга были проведены тесты в двух режимах (рис. 4): классическое обучение CNN и обучение с помощью СВР-метода. Для построения графиков были отобраны два файла с различным количеством объектов (35 000, 230 000), в качестве обучающей выборки использовались первые 1 000 объектов, прошедших процедуру классификации СВР-модулем.

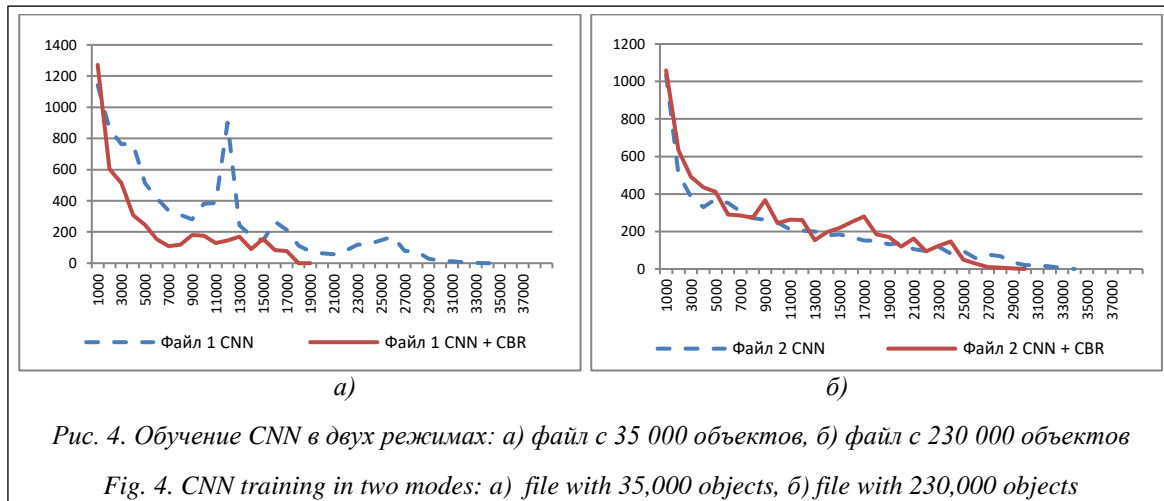
Для корректности выполнения тестирования оба метода начинали функционировать с одинаковыми значениями весовых коэффициентов CNN, шаг обучения изначально был равен 0,1. Обучение проходило на одних и тех же обучающих и проверочных выборках, критерий останова обучения – достижение порогового значения ошибки на всей обучающей выборке (0,1 %).

При анализе первого файла (рис. 4а) использование предложенного метода обучения CNN позволило достичь порогового значения ошибки на всей обучающей выборке (0,1 %) за значительно меньшее количество итераций (17 500) по сравнению с классическим методом (39 000).

На рисунке 4б хорошо иллюстрируется суть метода обучения CNN с помощью СВР-метода: шаг обучения калибруется для подбора оптимального значения. При работе с данными из второго файла (рис. 4б) предложенный метод также оказался предпочтительнее классического метода обучения CNN, так как потребовал меньшего количества итераций до достижения критерия останова обучения CNN.

### Заключение

В работе выполнена программная реализация расширения СВР-модуля с использованием сетей типа CNN для решения задачи классификации данных при малом объеме обучающей выборки. Разработанные программные средства способны работать в нескольких режимах: накопления обучающей выборки с использованием СВР-метода, обучения CNN с использованием СВР и в режиме функционирования обученной CNN. Благодаря интеграции СВР-метода и CNN удалось повысить эффективность решения задачи классификации данных.



Предложенный метод обучения CNN использует прецедентное представление итераций при обучении и позволяет обеспечить контроль за обучением сети (регулировку шага обучения). Каждое новое изменение шага вычисляется в соответствии с накопленным опытом, что увеличивает скорость обучения CNN.

Работа программных средств, реализованных на основе CBR-метода и CNN, была про-

верена на тестовых наборах данных для решения задачи классификации сигналов, полученных при акустико-эмиссионном мониторинге сложных технических объектов. Результаты вычислительных экспериментов подтвердили возможность повышения быстродействия алгоритма обучения CNN на основе CBR, благодаря чему можно сделать вывод о целесообразности использования предложенных решений.

*Работа выполнена при финансовой поддержке РФФИ, проекты №№ 18-01-00459, 18-29-03088, 20-07-00498, 20-57-00015.*

#### *Литература*

1. Варшавский П.Р., Алехин Р.В., Кожевников А.В. Разработка прецедентного модуля для идентификации сигналов при акустико-эмиссионном мониторинге сложных технических объектов // Программные продукты и системы. 2019. Т. 32. № 2. С. 207–213. DOI: 10.15827/0236-235X.126.207-213.
2. Варшавский П.Р., Барат В.А., Кожевников А.В. Прецедентный модуль для идентификации сигналов при акустико-эмиссионном мониторинге сложных технических объектов // Вестн. МЭИ. 2020. № 4. С. 122–128.
3. LeCun Y., Boser B., Denker J.S., Henderson D., Howard R.E., Hubbard W., Jackel L.D. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1989, vol. 1, no. 4, pp. 541–551.
4. Ciresan D.C., Meier U., Masci J., Luca M.J., Schmidhuber J. Flexible, high performance convolutional neural networks for image classification. *Proc. IJCAI*, 2011, pp. 1237–1242.
5. Богачев И.В., Левенец А.В., Чье Ен Ун. Применение искусственной нейронной сети для классификации телеметрических данных в системах сжатия // Информационно-управляющие системы. 2016. № 3. С. 2–7. DOI: 10.15217/issn1684-8853.2016.3.2.
6. Du K.-L. Clustering: a neural network approach. *Neural Networks*, 2010, vol. 23, pp. 89–107. DOI: 10.1016/j.neunet.2009.08.007.
7. Goodfellow I., Bengio Y., Courville A. *Deep Learning*. MIT Press, 2016, 800 p.
8. Еремеев А.П., Варшавский П.Р., Куриленко И.Е. Моделирование временных зависимостей в интеллектуальных системах поддержки принятия решений на основе прецедентов // International Journal Information Technologies and Knowledge. 2012. Т. 6. № 3. С. 227–239.
9. Рассел С., Норвиг П. Искусственный интеллект: современный подход; [пер. с англ.]. М.: Вильямс, 2006. 1408 с.
10. Albahari J., Albahari B. *C# 4.0 in a Nutshell: The Definitive Reference*. O'Reilly Media Publ., 2010, 1060 p.

## Implementation of data classification software based on convolutional neural networks and case-based reasoning approach

**P.R. Varshavskiy**<sup>1</sup>, Ph.D. (Engineering), Associate Professor, VarshavskyPR@mpei.ru  
**A.V. Kozhevnikov**<sup>1</sup>, Assistant, antoko@yandex.ru

<sup>1</sup> National Research University “Moscow Power Engineering Institute”,  
Department of Applied Mathematics and Artificial Intelligence, Moscow, 111250, Russian Federation

**Abstract.** This paper devotes to the implementation of software for data classification using case-based reasoning (CBR) and convolutional neural network technology (CNN). CBR-methods are widely used to find solutions to various problems based on accumulated experience, and CNN are successfully used in solving classification problems by isolating individual elements and forming high-level features using convolution kernels.

One of the necessary conditions for the success of solving the data classification problem is the presence of a correct training dataset. Unfortunately, this condition cannot always be fulfilled (for example, due to the complexity of the objects under consideration and lack of base information). Due to the ability to accumulate, use, and adapt existing experience, CBR-methods can be used to form a training dataset that can be further used by other methods to solve the data classification problem. Thus, the integration of CNN and CBR improves the efficiency of solving the data classification problem.

In addition, CBR-methods can be applied in areas with unpredictable behavior and can be trained in the process of functioning, for example, in the process of training neural networks. This paper proposes the CBR-method for CNN training, managing the process of training, and presentation of iterations of CNN training as a case. The selection of a training step based on precedents improves the performance of the neural network training algorithm.

Based on the proposed methods a neural network block using CNN for extending the capability of the CBR-system for data classification is implemented in MS Visual Studio in C# language. To evaluate the effectiveness of the solutions proposed in the work, computational experiments were performed on real data sets.

**Keywords:** precedent approach, convolutional neural network, neural network learning, data analysis, data classification.

**Acknowledgements.** This work was supported by RFBR, projects nos. 18-01-00459, 18-29-03088, 20-07-00498, 20-57-00015.

### References

1. Varshavskiy P.R., Alekhin R.V., Kozhevnikov A.V. Development of the case-based reasoning module for identification of acoustic-emission monitoring signals of complex technical objects. *Software & Systems*, 2019, vol. 32, no. 2, pp. 207–213 (in Russ.). DOI: 10.15827/0236-235X.126.207-213.
2. Varshavskiy P.R., Barat V.A., Kozhevnikov A.V. The case-based reasoning module for identification of acoustic-emission monitoring signals of complex technical objects. *Bull. of MPEI*, 2020, no. 4, pp. 122–128 (in Russ.).
3. LeCun Y., Boser B., Denker J.S., Henderson D., Howard R.E., Hubbard W., Jackel L.D. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1989, vol. 1, no. 4, pp. 541–551.
4. Ciresan D.C., Meier U., Masci J., Luca M.J., Schmidhuber J. Flexible, high performance convolutional neural networks for image classification. *Proc. IJCAI*, 2011, pp. 1237–1242.
5. Bogachev I.V., Levenets A.V., Chy E.U. Artificial neural networks in classifying telemetry data for compression systems. *Information and Control Systems*, vol. 3, 2016, pp. 2–7 (in Russ.). DOI: 10.15217/issn1684-8853.2016.3.2.
6. Du K.-L. Clustering: a neural network approach. *Neural Networks*, 2010, vol. 23, pp. 89–107. DOI: 10.1016/j.neunet.2009.08.007.
7. Goodfellow I., Bengio Y., Courville A. *Deep Learning*. MIT Press, 2016, 800 p.
8. Ereemeev A.P., Varshavskiy P.R., Kurilenko I.E. Modelling the time-dependent in intellectual systems of decision making support on the basis of precedents. *International Journal Information Technologies and Knowledge*, 2012, vol. 6, no. 3, pp. 227–239 (in Russ.).

9. Russell S., Norvig P. *Artificial Intelligence: A Modern Approach*. 2011, 937 p. DOI: 10.1016/j.artint.2011.01.005 (Russ. ed.: Moscow, 2006, 1408 p.).
10. Albahari J., Albahari B. *C# 4.0 in a Nutshell: The Definitive Reference*. O'Reilly media Publ., 2010, 1060 p.

#### Для цитирования

Варшавский П.Р., Кожевников А.В. Реализация программных средств для классификации данных на основе аппарата сверточных нейронных сетей и прецедентного подхода // Программные продукты и системы. 2020. Т. 33. № 4. С. 591–598. DOI: 10.15827/0236-235X.132.591-598.

#### For citation

Varshavskiy P.R., Kozhevnikov A.V. Implementation of data classification software based on convolutional neural networks and case-based reasoning approach. *Software & Systems*, 2020, vol. 33, no. 4, pp. 591–598 (in Russ.). DOI: 10.15827/0236-235X.132.591-598.