DOI: 10.12731/2227-930X-2023-13-4-142-158
УДК 004.8

Научная статья | Системный анализ, управление и обработка информации

ОПТИМИЗАЦИЯ НЕЙРОННЫХ СЕТЕЙ: МЕТОДЫ И ИХ СРАВНЕНИЕ НА ПРИМЕРЕ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ТЕКСТА

Ю.В. Торкунова, Д.В. Милованов

В результате исследования было разработано программное обеспечение, реализующее различные алгоритмы оптимизации нейронных сетей, позволившее провести их сравнительный анализ по качеству оптимизации. В статье подробно рассматриваются искусственные нейронные сети и методы их оптимизации: квантование, обрезка, дистилляция, разложение Такера. Описаны алгоритмы и инструменты оптимизации нейронных сетей, проведен сравнительный анализ различных методов, преимущества и недостатки, приведены расчетные значения и даны рекомендации по использованию каждого из методов. Оптимизация рассматривается на задаче классификации текстов, которые были предварительно подготовлены к обработке: извлечены признаки, выбраны и обучены модели, настроены параметры. Поставленная задача реализована при помощи технологий: языка программирования Python, фреймворка Pytorch, среды разработки Jupyter Notebook. Полученные результаты могут быть использованы в целях экономия вычислительных мощностей при сохранении качества распознавания и классификации.

Ключевые слова: *искусственные нейронные сети; оптимизация; сжатие и ускорение нейронных сетей; классификация текста; квантование; разложение Такера; дистилляция*

Для цитирования. *Торкунова Ю.В., Милованов Д.В. Оптимизация нейронных сетей: методы и их сравнение на примере интеллектуального анализа текста // International Journal of Advanced Studies. 2023. Т. 13, № 4. С. 142-158. DOI: 10.12731/2227-930X-2023-13-4-142-158*

Original article | System Analysis, Management and Information Processing

NEURAL NETWORKS OPTIMIZATION: METHODS AND THEIR COMPARISON BASED OFF TEXT INTELLECTUAL ANALYSIS

J.V. Torkunova, D.V. Milovanov

The research resulted in the development of software that implements various algorithms of neural networks optimization, which allowed to carry out their comparative analysis in terms of optimization quality. The article takes a detailed look at artificial neural networks and methods of their optimization: quantization, overcutting, distillation, Tucker's dissolution. Algorithms and optimization tools of neural networks were explained, as well as comparative analysis of different methods was conducted with their advantages and disadvantages listed. Calculation values were given as well as recommendations on how to execute each method. Optimization is studied by text classification performance: peculiarities were removed, models were chosen and taught, parameters were adjusted. The set task was completed with the use of the following technologies: Python programming language, Pytorch framework and Jupyter Notebook developing environment. The results that were acquired can be used to reduce the demand on computing power while preserving the same level of detection and classification abilities.

Keywords: *artificial neural networks; optimization; compression and accelerating of neural networks; text classification; quantization; Tucker's dissolution; distillation*

For citation. *Torkunova J.V., Milovanov D.V. Neural Networks Optimization: Methods and Their Comparison Based off Text Intellectual Analysis. International Journal of Advanced Studies, 2023, vol. 13, no. 4, pp. 142-158. DOI: 10.12731/2227-930X-2023-13-4-142-158*

Введение

Искусственные нейронные сети (ИНС) становятся все более актуальными в современном мире благодаря их способности учиться на больших и сложных наборах данных и делать точные прогнозы или классификации [1, 9]. ИНС – это тип алгоритма машинного обучения, который смоделирован по образцу структуры и функции человеческого мозга. Они состоят из слоев взаимосвязанных узлов или «нейронов», которые обрабатывают и передают информацию.

Одним из ключевых преимуществ ИНС является их способность учиться на данных. Они могут автоматически выявлять закономерности и взаимосвязи в больших и сложных наборах данных, которые находятся за пределами человеческих возможностей. Это делает их особенно полезными в таких областях, как распознавание изображений, распознавание речи и обработка естественного языка, где входные данные не структурированы и их трудно обрабатывать традиционными методами [6, 12].

Еще одним преимуществом ИНС является их гибкость. Нейронные сети можно спроектировать и обучить для решения широкого круга задач, таких как регрессия, классификация и кластеризация. Их также можно применять в сценариях обучения без учителя и с учителем. Эта гибкость делает их универсальным инструментом, который можно адаптировать к различным областям и приложениям. Кроме того, ИНС можно использовать в сочетании с другими методами машинного обучения, такими как обучение с подкреплением, для решения еще более сложных задач.

Материалы и методы

По мере того как нейронные сети становятся все более сложными и применяются к большим наборам данных, растет потребность в эффективных и быстрых моделях. Один из подходов к решению этой задачи – сжатие и ускорение нейронных сетей.

Сжатие нейронных сетей относится к процессу уменьшения размера модели при сохранении или улучшении ее производительности. Это может быть достигнуто с помощью различных методов, таких как обрезка, квантование и дистилляция знаний. Сокращение включает в себя удаление ненужных соединений или узлов в сети, в то время как квантование включает в себя уменьшение количества битов, используемых для представления весов и активаций сети. Дистилляция знаний включает в себя обучение небольшой сети, имитирующей поведение большей сети.

Одним из ключевых преимуществ сжатия нейронных сетей является снижение требований к хранилищу и памяти, что может привести к более быстрому выводу и снижению энергопотребления. Кроме того, сжатые модели легче развертывать на устройствах с ограниченными ресурсами, таких как мобильные телефоны и встроенные системы. В некоторых случаях сжатые модели могут даже превзойти свои более крупные аналоги благодаря улучшенному обобщению и уменьшению переобучения [4].

Актуальность сжатия и ускорения нейронных сетей очевидна в их потенциале для повышения эффективности, скорости и производительности моделей. Разработка методов сжатия и ускорения необходима для реализации всего потенциала нейронных сетей и обеспечения того, чтобы их влияние было полезным для общества.

Однако есть и проблемы, связанные со сжатием и ускорением нейронных сетей. Одной из проблем является компромисс между размером модели и производительностью. Хотя сжатые и ускоренные модели могут быть меньше и быстрее, они также могут оказывать негативное влияние на точность или способности к

обобщению. Уравновешивание этого компромисса является ключевым фактором при применении этих методов [7, 8].

Другой проблемой является сложность реализации методов сжатия и ускорения. Различные методы могут потребовать специальных знаний и опыта, а оптимальный подход может различаться в зависимости от конкретного варианта использования и набора данных. Кроме того, использование аппаратного ускорения может потребовать инвестиций в специализированное оборудование и инфраструктуру.

В целом актуальность сжатия и ускорения нейронных сетей значительна, так как это позволяет использовать нейронные сети в более широком спектре приложений, включая те, которые требуют эффективных и быстрых моделей. Несмотря на проблемы, связанные с этими методами, продолжение исследований и разработок в этой области имеет важное значение для реализации всего потенциала нейронных сетей в различных областях, а также для обеспечения того, чтобы их влияние было этичным и полезным для общества. Существует несколько методов оптимизации нейронных сетей. Рассмотрим их.

Квантование – это процесс снижения точности весов и активаций нейронных сетей для уменьшения их требований к вычислениям и памяти. Квантование можно применять к весам и активациям нейронных сетей. Квантование весов включает представление весов с меньшим количеством битов, в то время как квантование активации включает снижение точности значений активации [13].

Данный метод является эффективным методом снижения вычислительной сложности и требований к памяти нейронных сетей. Например, 8-битное квантование с фиксированной точкой может снизить требования к памяти сверточных нейронных сетей до 4 раз без существенного снижения точности. Точно так же квантование по степени двойки может обеспечить двукратное ускорение времени вывода с незначительной потерей точности.

Важно отметить, что популярные инструменты для разработки нейронных сетей часто не поддерживают использование графического процессора для ускорения вычислений нейронных сетей, которые были квантованы. Хотя, в зависимости от используемой реализации квантования, допускается возможность использования графического процессора с определёнными ограничениями и нюансами.

Эффективность квантования зависит от архитектуры нейронной сети, набора данных и конкретного используемого метода квантования. Например, квантование может иметь большее влияние на точность для сложных архитектур, таких как рекуррентные нейронные сети, чем для простых архитектур, таких как нейронные сети с прямой связью.

Неструктурированная обрезка весов включает в себя удаление отдельных весов с небольшими величинами или их обнуление, что уменьшает количество параметров в сети. Этот тип обрезки может выполняться итеративно во время обучения или после обучения сети. Итеративное сокращение весов включает в себя сокращение небольшого процента весов на каждой итерации, обучения сети с оставшимися весами и повторение процесса до тех пор, пока не будет достигнут желаемый уровень разреженности. Сокращение веса после обучения включает в себя удаление определенного процента весов из полностью обученной сети и последующее обучение оставшихся весов для восстановления производительности.

Обрезку можно рассматривать как форму регуляризации, которая представляет собой метод, используемый для предотвращения переобучения в моделях машинного обучения. Регуляризация включает в себя добавление штрафного члена к функции потерь, чтобы побудить модель иметь меньшие веса или меньше параметров. Обрезка достигает аналогичного эффекта за счет уменьшения количества соединений и/или нейронов в сети, что может предотвратить переобучение и повысить способность к обобщению

Кроме того, обрезку также можно использовать в сочетании с другими методами регуляризации, такими как weight decay, dropout и batch normalization. Weight decay включает в себя добавление штрафного члена к функции потерь, которая поощряет малые веса, dropout случайным образом отбрасывает некоторые нейроны во время обучения, чтобы предотвратить переобучение, a batch normalization нормализует активацию каждого слоя, чтобы предотвратить проблемы с исчезновением и взрывом градиента. Вместе с сокращением эти методы могут еще больше повысить производительность и надежность нейронных сетей.

Принимая во внимание приведённые выше аргументы, можно говорить о том, что обрезка – это хороший метод уменьшения размера, сложности и вычислительной стоимости нейронных сетей при сохранении их производительности. Сокращение может выполняться итеративно во время обучения или после обучения сети и может включать удаление весов, нейронов или слоев из сети. Критерии обрезки могут быть основаны на величинах или чувствительности весов, нейронов или фильтров. Обрезку также можно использовать в сочетании с другими методами регуляризации для дальнейшего повышения производительности и надежности нейронных сетей. Наконец, сокращение может позволить развертывание нейронных сетей на периферийных устройствах, что может повысить эффективность и скорость отклика.

Дистилляция, также известная как дистилляция знаний, представляет собой метод передачи знаний из большой сложной нейронной сети, известной как сеть учитель, в меньшую и более простую нейронную сеть, известную как сеть ученик. Цель дистилляции – сохранить производительность сети учителя при уменьшении размера и вычислительной стоимости сети ученика [14].

Основная идея дистилляции состоит в том, чтобы научить сеть ученика имитировать поведение сети учителя. Это достигается путем обучения сети ученика таким образом, чтобы она одновременно решала две задачи, первой задачей будет являться пред-

сказанием целевой метки набора данных и функцией ошибки может быть перекрёстной-энтропией между правильным ответом и предсказанием модели ученика, вторая задача заключается в минимизации разницы между ответом модели ученика и учителя. Таким образом нейронной сети будет необходимо решать одновременно две задачи, предсказание правильного ответа и повторения предсказания модели учителя [15].

Дистилляция имеет ряд преимуществ перед традиционными методами обучения. Во-первых, его можно использовать для уменьшения размера и вычислительных затрат нейронных сетей при сохранении их производительности. Во-вторых, его можно использовать для повышения производительности небольших нейронных сетей, у которых может не хватить мощности для захвата всей соответствующей информации в данных. В-третьих, его можно использовать для передачи знаний между различными типами нейронных сетей, например, между сверточной нейронной сетью и рекуррентной нейронной сетью [10, 11].

Таким образом, принимая во внимание приведённые выше аргументы, стоит отметить, что дистилляция — это мощная и современная техника для передачи знаний из большой сложной нейронной сети в меньшую и более простую нейронную сеть. Качество итоговой модели ученика зависит от сложности поставленной задачи, количества параметров и качества модели учителя. Задача нахождения баланса между качеством и числом параметров сети особо важна для этого метода. Потенциально этот метод может дать впечатлительные результаты.

Сжатие нейронной сети с помощью матричной декомпозиции – это метод, который включает в себя разбиение весовых матриц нейронной сети на большое число более мелких матриц. Это может помочь уменьшить количество параметров в сети и сделать ее более эффективной с точки зрения вычислений [8, 9].

Использование методов для сжатия и ускорения работы нейронных сетей применялась при решении задачи бинарной клас-

сификации текстов с использование нейронной сети с архитектурой BeRT на одинаковом наборе данных для каждого из методов оптимизации.

Классификация текста – это задача обработки естественного языка (NLP), которая включает в себя классификацию текстовых документов по заранее определенным категориям. Эта задача используется в различных приложениях, таких как анализ настроений, фильтрация спама, моделирование тем и многое другое.

Существует несколько способов решения задач классификации текста, в том числе:

Методы на основе правил. Этот подход включает создание набора правил или эвристик, которые можно использовать для классификации текста. Например, основанный на правилах метод фильтрации спама может включать поиск определенных ключевых слов или шаблонов в тексте.

Методы машинного обучения: этот подход включает в себя обучение алгоритма машинного обучения для классификации текста на основе признаков, извлеченных из текста. Примеры алгоритмов машинного обучения, используемых для классификации текста, включают деревья решений, метод наивного Байеса, машины опорных векторов (SVM) и модели глубокого обучения, такие как рекуррентные нейронные сети (RNN), нейронные сети с механизмом само-внимания (Transformer).

Гибридные методы: этот подход сочетает в себе методы на основе правил и машинного обучения для повышения точности задачи классификации. Например, гибридный метод может использовать метод на основе правил для фильтрации очевидных случаев спама, а затем использовать алгоритм машинного обучения для классификации оставшихся сообщений.

Рассмотрим данные методы применительно к проблеме интеллектуального анализа текста. Очевидно, что прежде чем обрабатывать тексты, массивы текстовых данных необходимо подготовить должным образом и выполнить ряд работ, а именно:

- предварительную обработку данных: очистку и подготовку данных для анализа, в частности, удаление стоп-слов, выделение текста или лемматизацию текста, а также преобразование текста в стандартный формат.
- извлечение признаков: выбор наиболее релевантных признаков из текста, которые можно использовать для классификации, в частности, частоту слов, n-граммы и моделирование темы.
- выбор модели и обучение: выбор подходящего алгоритма машинного обучения и его обучение на размеченном наборе данных. Производительность модели можно оценить с помощью таких показателей, как точность, полнота и оценка F1.
- настройка параметров: точная настройку параметров алгоритма машинного обучения для повышения его производительности на тестовых данных [2, 3].

Для реализации и обучения нейронных сетей использовался популярный в настоящее время фреймворк - PyTorch. В данной реализации представлены как базовые слои нейронных сетей, так и современные архитектуры. Для обучения можно использовать множество готовых функций потерь, для решения различных задач и оптимизаторы, используемые для оптимального обновления весовых коэффициентов нейронных сетей. Обработка и приведение данных в необходимый формат происходит также при помощи данного фреймворка.

Перед началом применения методов оптимизации сетей, необходимо обучить базовую версию модели, с которой будет сравниваться качество и быстродействие оптимизированных нейронных сетей. Качество будет измеряться значением F1 метрики. Также данные будут разбиты на две части, на одной части данных будет происходить обучение модели, а на второй измеряться её качество. Набор для обучения будет больше, чем для тестирования сети. Соотношение размера набора для обучения к набору для тестирования составляет 4 к 1.

Для обновления параметров нейронной сети, а также её обучения используется функция ошибки потерь, это ошибки между прогнозируемым выходом сети и истинным выходом. В задачах классификации цель состоит в том, чтобы предсказать метку класса для данного входа.

Для обучения выбранной модели будет использоваться одна из самых широко используемых функций потерь для задачи классификации. Перекрестная энтропийная потеря: это наиболее часто используемая функция потерь для задач классификации. Она измеряет разницу между предсказанным распределением вероятностей и истинным распределением вероятностей.

Важно выбрать подходящую функцию потерь для задачи классификации, исходя из характера данных и желаемого результата. Выбор функции потерь влияет на оптимизацию нейронной сети и может повлиять на точность прогнозов.

Вся разработка была произведена в среде – Jupyter Notebook. Jupyter Notebook – это интерактивная вычислительная среда, которая позволяет пользователям создавать и обмениваться документами, содержащими живой код, уравнения, визуализации и описательный текст.

Выбор аргументов для обучения модели осуществляется при помощи TrainingArgumets. Оно необходимо для создания функционала, отвечающий за обучение нейронной сети. В Trainer передаём нейронную сеть, аргументы обучения и данные. Чтобы запустить процесс обучения необходимо вызвать функцию train() у объекта trainer.

После обучения модели и оценки её качества и быстродействия можно переходить к оптимизации этой нейронной сети. После каждого метода оптимизации оценка качества и быстродействия будет происходить на тех же данных, что и для исходной модели.

Результаты

Для каждого из ранее описанного метода было замерено качество решения задачи бинарной классификации, размер модели в МБ, а также её быстродействие на центральном и графическом

процессорах. По возможности, часть методов комбинировалась друг с другом. Комбинация методов происходила последовательно, в определенном порядке, ведь некоторые методы лучше применить только после обучения модели. Результаты работы методов оптимизации отображены в Таблице 1. Квантование происходило в формат qint8, из-за чего использование графического процессора недоступно, поэтому для данного метода нет данных по быстродействию с применением графического процессора

Таблица 1.

Сравнительный анализ результатов применения методов оптимизации

Метод	Размер модели (МБ)	Ошибка F1	Скорость на графическом процессоре, с.	Скорость на центральном процессоре, с.
Без применения оптимизации	711	0,966	0,036	1,51
Квантование линейных слоёв	454	0,966	-	1,18
Квантование линейных слоёв и слоя векторного представления слов	179	0,966	-	0,13
Разложение Такера	385	0,966	0,036	0,78
Разложение Такера и квантование слоя векторного представления слов	109	0,967	-	0,817
Дистилляция	10,3	0,952	0,005	0,12
Квантование дистиллированной модели	2,68	0,952	-	0,105
Разложение Такера на дистиллированной модели	3,43	0,951	0,02	0,11
Разложение Такера и квантование слоя векторного представления слов для дистиллированной модели	1,69	0,958	-	0,1
Обрезка	598	0,975	0,029	1,15

Иногда, после применения метода оптимизации, улучшалось качество модели, это происходило по случайности. Причиной

этому может быть малый объем набора данных. Результаты методов могут отличаться при использовании различных технических устройств, а также других наборов данных и архитектур нейронных сетей.

Данная таблица носит сравнительный характер для методов между собой в одинаковых условиях.

Обсуждение результатов

В данной статье были рассмотрены и реализованы методы оптимизации нейронных сетей, которые позволяют расширить применение нейронных сетей. Эта задача актуальна и очень важна в настоящее время, нейронные сети применяются многими компаниями, вне зависимости от её размера и использование подобных методов позволяет сократить ресурсы, необходимые для эксплуатации нейронных сетей. Каждый из этих методов имеет свои преимущества и недостатки. Возможность комбинации некоторых методов имеют большое влияние на итоговый размер сети и её быстродействие.

Заключение

Результатом выполнения данной работы является реализованные методы оптимизации нейронных сетей, а также сравнительный анализ этих методов. Реализация происходила на языке программирования Python и в основном использовались следующие пакеты: pytorch, transformers, pandas.

Список литературы

1. Аветисян Т. В., Львович Я. Е., Преображенский А. П. Разработка подсистемы распознания сигналов сложной формы // International Journal of Advanced Studies. 2023. Т. 13, № 1. С. 102-114. https://doi.org/10.12731/2227-930X-2023-13-1-102-114
2. Акжолов Р.К., Верига А.В. Предобработка текста для решения задач NLP // Вестник науки. 2020. № 3 (24). С. 66-68.

3. Ахметзянова К.Р., Тур. А.И., Кокоулин А.Н. Оптимизация вычислений нейронной сети // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. 2020. № 36. С. 117-130. https://doi.org/10.15593/2224-9397/2020.4.07
4. Каширина И. Л., Демченко М. В. Исследование и сравнительный анализ методов оптимизации, используемых при обучении нейронных сетей // Вестник ВГУ, серия: Системный анализ и информационные технологии, 2018, № 4. С.123-132.
5. Копырин А. С., Макарова И. Л. Алгоритм препроцессинга и унификации временных рядов на основе машинного обучения для структурирования данных // Программные системы и вычислительные методы. 2020. № 3. С. 40-50. https://doi.org/10.7256/2454-0714.2020.3.33958
6. Осовский С. Нейронные сети для обработки информации. М.: Горячая линия. Телеком. 2019. 448 с.
7. Романов Д.Е. Нейронные сети обратного распространения ошибки // Инженерный вестник Дона. 2009. № 3 . С. 19-24.
8. Созыкин А.В. Обзор методов обучения глубоких нейронных сетей // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2017. № 3 (6). С. 28-59.
9. Торкунова Ю.В., Коростелева Д.М., Кривоногова А.Е. Формирование цифровых навыков в электронной информационно-образовательной среде с использованием нейросетевых технологий // Современное педагогическое образование. 2020. №5. С. 107-110.
10. Черкасова И.С. Оптимизация гиперпараметров нейронной сети и снижение вычислительных затрат // E-Scio. 2022. https://e-scio.ru/wp-content/uploads/2022/03/%D0%A7%D0%B5%D1%80%D0%BA%D0%B0%D1%81%D0%BE%D0%B2%D0%B0-%D0%98.-%D0%A1.pdf (дата обращения: 13.04.2023).
11. Ященко А.В., Беликов А.В., Петерсон М.В. Дистилляция нейросетевых моделей для детектирования и описания ключевых точек изображений // Научно-технический вестник информационных технологий, механики и оптики. 2020. № 3. С. 402-409.

12. A White Paper on Neural Network Quantization. https://doi.org/10.48550/arXiv.2106.08295
13. Distilling Task-Specific Knowledge from BERT into Simple Neural Networks. https://doi.org/10.48550/arXiv.1903.12136
14. Majid Janzamin, Rong Ge, Jean Kossaifi and Anima Anandkumar. Spectral Learning on Matrices and Tensors // Foundations and Trends R in Machine Learning, 2019. Vol. 12, No. 5-6. P. 393–536. https://doi.org/10.1561/2200000057
15. Tensor Networks for Latent Variable Analysis. Part I: Algorithms for Tensor Train Decomposition. https://arxiv.org/pdf/1609.09230.pdf (дата обращения: 20.05.2023)

References

1. Avetisyan T. V., L'vovich Ya. E. *International Journal of Advanced Studies*, 2023, vol. 13, no. 1, pp. 102-114. https://doi.org/10.12731/2227-930X-2023-13-1-102-114
2. Akzholov R.K., Veriga A.V. *Vestnik nauki,* 2020, no. 3 (24), pp. 66-68.
3. Akhmetzyanova K.R., Tur. A.I., Kokoulin A.N. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotekhnika, informatsionnye tekhnologii, sistemy upravleniya,* 2020, no. 36, pp. 117-130. https://doi.org/10.15593/2224-9397/2020.4.07
4. Kashirina I. L., Demchenko M. V. *Vestnik VGU, seriya: Sistemnyy analiz i informatsionnye tekhnologii*, 2018, no. 4, pp. 123-132.
5. Kopyrin A. S., Makarova I. L. *Programmnye sistemy i vychislitel'nye metody,* 2020, no. 3, pp. 40-50. https://doi.org/10.7256/2454-0714.2020.3.33958
6. Osovskiy S. *Neyronnye seti dlya obrabotki informatsii* [Neural networks for information processing]. Moscow: Hot Line. Telecom. 2019, 448 p.
7. Romanov D.E. *Inzhenernyy vestnik Dona,* 2009, no. 3, pp. 19-24.
8. Sozykin A.V. *Vestnik YuUrGU. Seriya: Vychislitel'naya matematika i informatika,* 2017, no. 3 (6), pp. 28-59.
9. Torkunova Yu.V., Korosteleva D.M., Krivonogova A.E. *Sovremennoe pedagogicheskoe obrazovanie,* 2020, no. 5, pp. 107-110.

10. Cherkasova I.S. *E-Scio,* 2022. https://e-scio.ru/wp-content/uploads/2022/03/%D0%A7%D0%B5%D1%80%D0%BA%D0%B0%D1%81%D0%BE%D0%B2%D0%B0-%D0%98.-%D0%A1.pdf
11. Yashchenko A.V., Belikov A.V., Peterson M.V. *Nauchno-tekhnicheskiy vestnik informatsionnykh tekhnologiy, mekhaniki i optiki,* 2020, no. 3, pp. 402-409.
12. A White Paper on Neural Network Quantization. https://doi.org/10.48550/arXiv.2106.08295
13. Distilling Task-Specific Knowledge from BERT into Simple Neural Networks. https://doi.org/10.48550/arXiv.1903.12136
14. Majid Janzamin, Rong Ge, Jean Kossaifi and Anima Anandkumar. Spectral Learning on Matrices and Tensors. *Foundations and Trends R in Machine Learning*, 2019, vol. 12, no. 5-6, pp. 393–536. https://doi.org/10.1561/2200000057
15. Tensor Networks for Latent Variable Analysis. Part I: Algorithms for Tensor Train Decomposition. https://arxiv.org/pdf/1609.09230.pdf

ДАННЫЕ ОБ АВТОРАХ

Торкунова Юлия Владимировна, профессор кафедры «Информационные технологии и интеллектуальные системы», доктор педагогических наук

Казанский государственный энергетический университет; Сочинский государственный университет

ул. Красносельская, 51, г. Казань, Республика Татарстан, 420066, Российская Федерация; ул. Пластунская, 94, г. Сочи, Краснодарский край, 354000, Российская Федерация

torkynova@mail.ru

Милованов Данила Владиславович, магистр

Казанский государственный энергетический университет

ул. Красносельская, 51, г. Казань, Республика Татарстан, 420066, Российская Федерация

studydmk@gmail.com

DATA ABOUT THE AUTHORS

Julia V. Torkunova, Professor of the Department of Information Technologies and Intelligent Systems, Doctor of Pedagogical Sciences

Kazan State Power Engineering University; Sochi State University

51, Krasnoselskaya Str., Kazan, Republic of Tatarstan, 420066, Russian Federation; 94, Plastunskaya Str., Sochi, Krasnodar region, 354000, Russian Federation

torkynova@mail.ru

SPIN-code: 7422-4238,

ORCID: https://orcid.org/0000-0001-7642-6663

Danila V. Milovanov, Magister

Kazan State Power Engineering University

51, Krasnoselskaya Str., Kazan, Republic of Tatarstan, 420066, Russian Federation

studydmk@gmail.com

Поступила 13.11.2023
После рецензирования 28.11.2023
Принята 02.12.2023

Received 13.11.2023
Revised 28.11.2023
Accepted 02.12.2023