УДК 378

С. Д. Коткин

S. D. Kotkin

Коткин Сергей Дмитриевич, к. п. н., доцент кафедры ИОТД, Кузбасский гуманитарно-педагогический институт Кемеровского государственного университета, г. Новокузнецк, Россия.

Kotkin Sergey Dmitrievich, Ph.D., Associate Professor, Kuzbass Humanitarian Pedagogical Institute Kemerovo State University, Novokuznetsk, Russia.

ОСОБЕННОСТИ ОБУЧЕНИЯ СТУДЕНТОВ ПЕДАГОГИЧЕСКИХ СПЕЦИАЛЬНОСТЕЙ СОЗДАНИЮ НЕЙРОННЫХ СЕТЕЙ

FEATURES OF TRAINING STUDENTS OF PEDAGOGICAL SPECIALTIES IN CREATION OF NEURAL NETWORKS

Аннотация. *В статье рассматривается организация изучения нейронных сетей студентами педагогических специальностей. Даются предложения по использованию в учебном процессе таких инструментальных средств как язык программирования Python и библиотека Keras.*

Annotation. *The article deals with the organization of the study of neural networks by students of pedagogical specialties. Suggestions are given on the use of such tools in the educational process as the Python programming language and the Keras library.*

Ключевые слова: *педагогическое образование, системы искусственного интеллекта, машинное обучение, нейронные сети.*

Keywords: *teacher education, artificial intelligence systems, machine learning, neural networks.*

Одной из тенденций развития современного общества является начавшийся переход на пятый технологический уклад, включая повсеместное внедрение достижений микроэлектронной промышленности, а также цифровизация процессов управления, как на уровне отдельных производств, так и на уровне государств. Так в Российской федерации к сферам осуществления цифровой трансформации относится управление экономикой, государственным имуществом, надзорная деятельность, государственные и муниципальные услуги, а также государственная безопасность. При этом к проблемам осуществления данного процесса относится, в том числе, недостаточная достоверность необходимых данных [1]. Ответом на данный вызов являет предложение Министерства науки и высшего образования РФ включить в основные образовательные программы высшего образования по всем специальностям и направлениям подготовки модуля «Системы искусственного интеллекта» [2]. Объем данного модуля составляет 108 часов (3 з.е.) в базовой части и 36 часов (1 з.е.) в вариативной части. Целью освоения модуля заявлено «овладение студентами основными методами теории интеллектуальных систем, приобретение навыков по использованию интеллектуальных систем, изучение основных методов представления знаний и моделирования рассуждений» [2]. К темам, изучаемым в модуле, относятся инженерия знаний, нечеткие множества, экспертные системы, нейронные сети, а также их применение в построении интеллектуальных систем и интеллектуальных интерфейсов. Таким образом, все современные студенты, независимо от профиля подготовки должны быть готовы к пониманию принципов работы систем искусственного интеллекта.

Среди специальностей направления 44.03 есть как специальности, где нейронные сети используются исключительно как прикладной инструмент для анализа данных, так и специальности, предполагающие владением принципами создания подобных инструментов, в том числе для обучения школьников в ходе проведения занятий в предпрофильных и профильных классах и при организации факультативных уроков и занятий в дополнительном образовании, а также в ходе подготовки студентов СПО. И если для первой группы педагогических специальностей достаточно иметь представление о возможностях нейронных сетей: круге решаемых с их помощью задач, свойственные им ограничения, преимущества и недостатки их использования в сравнении с прочими методами решения подобных задач, то студентам второй группы специальностей необходимо иметь представления об устройстве нейронных сетей, о выборе их архитектуры, алгоритмах начального выбора и изменения гиперпараметров и т.д.

С этой целью студентам необходимо получить как теоретические знания, так и практический опыт в области использования нейронных сетей для решения некоторого набора задач. Что это могут быть за задачи? Возможным направлением внедрения изучения нейронных сетей в школах могут быть такие, которые способствуют начавшимся в последние годы в России процессам реиндустриализации, импортозамещения и создания высокотехнологичных производств. Профориентации школьников в данных направлениях способствуют такие области, где могут использоваться нейронные сети, как робототехника, системы автоматического управления, включая системы интернета вещей и умного дома, системы машинного зрения и дополненной реальности. Во всех этих областях применение более традиционных систем искусственного интеллекта, основанных на формальных моделях: логических, продукционных, семантических и фреймовых, затруднено, поскольку требует участия человека как на этапе создания модели так и, зачастую, на этапе создания представления. Поэтому, хоть и традиционные системы искусственного интеллекта и машинного обучения давно и успешно используются для работы с абстрактными задачами в экспертных системах, эксплуатирующихся специалистами в предметной области (которые как раз и отвечают за создание представления ситуации), в системах автоматического вывода и доказательства теорем, в программах для логических игр [3, 4], к решению задач, в которых данные поступают непосредственно из окружающего мира, без предварительной обработки их человеком, они малопригодны.

В отличие от подобных систем, нейронные сети способны самообучаться, самостоятельно создавая подходящее представление задачи, что позволяет использовать их в автоматических системах и автоматизированных системах, которые не требуют для своего функционирования эксперта в предметной области, способного конструировать абстрактные описания задач [5].

Хотя нейронные сети имеют уже семидесятилетнюю историю, переживали и всплески и спад интереса к ним, практическое их применение стало возможным лишь в десятых годах этого века, факторами чего явились рост производительности вычислительной техники при увеличении её доступности, и, что даже более важно, создание массивов цифровых данных, пригодных для обучения нейронных сетей и исследования их архитектуры. Вместе с этим стал расширяться спектр существующих архитектур нейронных сетей и, как следствие, спектр задач, которые доступны для решения с их помощью. Если изначально нейронные сети применялись преимущественно для задач классификации, кластеризации, снижения размерности, регрессии, то есть тех, для которых существуют и полностью детерминированные алгоритмы решений, то нынешние нейронные сети пригодны и для решения задач, для которых традиционные подходы до сих пор оставались малопригодными. В частности, благодаря нейронным сетям значительно увеличилась точность задач классификации образов, улучшилось качество машинного перевода, появились системы, генерирующие тексты на основе начального фрагмента, синтезирующие изображения или видео путем достраивания исходных изображений, озвучивающие текст голосом, мало отличимым от голоса живого человека, а также системы, решающие задачи, до сих пор не имевшие приемлемых, по вычислительным затратам, решений: фолдинг белков [6], игра Го, или расчет перемещения роботов-манипуляторов с помощью микроконтроллеров [7].

В создании нейронных сетей большую роль играют обучающие данные, которые могут браться как из открытых, так и из коммерческих источников. Однако перед использованием в обучении их зачастую необходимо особым образом подготовить, привести к единому формату, иногда предварительно классифицировать. Эти операции требуют участия человека. При этом трудозатраты на подготовку данных растут с их объемом. Но большой объем обучающих данных, как правило, увеличивает точность работу нейронной сети. Поэтому данный этап часто является самым дорогим в создании нейронной сети, а методы, позволяющие увеличить набор обучающих данных без дополнительных трудозатрат, зачастую открывают для нейронных сетей ранее недоступные классы задач.

Этап выбора архитектуры и гиперпараметров нейронной сети до сих пор остается больше вопросом мастерства. И хотя для некоторых классов задач существуют рекомендации, основанные на практическом опыте, часто именно структура нейронной сети является тем know how, который определяет её перспективы. При этом в настоящее время появились инструменты автоматизации поиска лучшей структуры нейронной сети. И хотя они позволяют с помощью различных алгоритмов оптимизации сократить время поиска подходящих гиперпараметров, начальный выбор архитектуры всё равно оказывает решающее влияния на результат. А так как этап обучения нейронной сети является очень требовательным к вычислительным ресурсам, на котором в итеративном процессе обратного распространения ошибки при обработке обучающих данных определяются весовые коэффициенты нейронов, определить правильность выбора архитектуры сети удается далеко не сразу. Из-за этого процесс обучения приходится повторять, от чего кратно возрастают вычислительные затраты. И при этом гарантий того, что определенную задачу удастся решить, при этом нет. Однако для типовых классов задач существует множество хорошо изученных архитектур с описанными преимуществами и недостатками, рекомендациями к начальному выбору гиперпараметров и объему обучающих выборок, что позволяет для таких задач получать с помощью нейронных сетей результаты за предсказуемое время.

При обучении студентов созданию и применению нейронных сетей можно ориентироваться на хорошо изученные в настоящее время архитектуры, делая упор на изучение инструментальных средств и на методических аспектах обучения основам работы с нейронными сетями школьников и студентов СПО.

Несмотря на то, что среди современных инструментов работы с нейронными сетями есть различные языки программирования и различные предметные библиотеки, зачастую имеющие программные интерфейсы для нескольких языков, наиболее подходящим в настоящее время является язык Python, используемый совместно с библиотекой Keras. Выбор данного языка обусловлен несколькими причинами. По многим рейтингам [8, 9, 10] данный язык в последние годы входит в пятерку самых востребованных работодателями. Данный язык имеет развитый синтаксис и поддерживает различные структуры данных, но, в отличие от таких языков, как Java или C++ имеет невысокий порог вхождения, так как изначально разрабатывался как скриптовый язык для решения небольших задач. Несмотря на свою невысокую производительность он пригоден для работы с нейронными сетями, так как для этого в нём используются библиотеки, написанные на высокопроизводительных языках C и C++ и оптимизированные для различных процессоров и различных графических ускорителей. Также этот язык имеет множество библиотек для решения прочих типов задач машинного обучения, работы с данными в различных форматах и библиотек визуализации данных. Также для данного языка существует множество интегрированных средств разработки с обладающими достаточной для использования в образовательном процессе функциональностью бесплатными версиями, в том числе такая IDE, как Jupiter Notebook, позволяющая в одном документе объединять программный код, результаты его работы и форматированные текстовые описания, что позволяет, помимо решения задач, создавать и обучающие материалы.

Наиболее удобно в учебных целях использовать Python совместно с библиотекой Keras [11, 12]. Она образует промежуточный слой между инетпретатором Python и библиотеками, непосредственно выполняющими вычисления. В предыдущих версиях она поддерживала множество библиотек, таких как TensorFlow, Microsoft Cognitive Toolkit, Theano и т.д. В последней версии в ней осталась поддержка только TensorFlow – на сегодняшний день самой распространенной библиотеки для работы с нейронными сетями. Функции Keras – скрыть от пользователя сугубо технические детали реализации нейронных сетей, оставив возможность создавать любые доступные в рамках поддерживаемых ею библиотек архитектуры.

Предложенное сочетание языка программирования, библиотек и инструментальных средств позволит как организовать обучение работе с нейронными сетями, так и выполнять исследовательские проекты, создавая новые архитектуры нейронных сетей и решая новые для данной области классы задач.

Список литературы

1. Российская Федерация. Стратегическое направление в области цифровой трансформации государственного управления (утверждено распоряжением Правительства Российской Федерации от 22 октября 2021 г. № 2998-р) [Электронный ресурс]. – URL : http://static.government.ru/media/files/

d3uclO4ZFGNKmxCPBXbL4OaMPALluGdQ.pdf (дата обращения : 09.11.2021).
2. Российская Федерация. Письмо Министерства науки и высшего образования РФ от 2 июля 2021 г. N МН-5/2657 «О направлении информации» [Электронный ресурс]. – URL : http://fgosvo.ru/uploadfiles/metod/Ps_MON_5_2657__02072021.pdf (дата обращения: 18.10.2021).
3. Осуга, С. Обработка знаний [Текст]. / С. Осуга, пер. с япон. – М. : Мир, 1989. – 293 с.
4. Джексон, П. Введение в экспертные системы [Текст]. / П. Джексон, пер. с англ. : Уч. пос. – М. : Издательский дом «Вильямс», 2001. – 624 с. : ил.
5. Гудфеллоу, Я. Глубокое обучение [Текст]. / Я. Гудфеллоу, И. Бенджио, А. Курвилль, пер. с англ. А. А. Слинкина. – 2-е изд., испр. – М. : ДМК Пресс, 2018. – 652 с. : цв. ил.
6. Crawford, J. What is the «protein folding problem»? A brief explanation [Электронный ресурс]. / J. Crawford // The Roots of Progress, 2020 – URL : https://rootsofprogress.org/alphafold-protein-folding-explainer (дата обращения : 16.12.2020).
7. Кожевников, М. М. Планирование траекторий промышленных роботов-манипуляторов на основе нейронных сетей [Электронный ресурс] / М. М. Кожевников, А. П. Пашкевич, О. А. Чумаков. // Доклады БГУИР, 2010. – № 4 (50). – URL : https://cyberleninka.ru/article/n/planirovanie-traektoriy-promyshlennyh-robotov-manipulyatorov-na-osnove-neyronnyh-setey (дата обращения : 28.11.2020).
8. TIOBE Index [Электронный ресурс]. – URL : https://www.tiobe.com/tiobe-index/ (дата обращения : 02.11.2021).
9. IEEE Spectrum Top Programming Languages [Электронный ресурс]. – URL : https://spectrum.ieee.org/top-programming-languages/ (дата обращения : 02.11.2021).
10. 2020 Developer Survey [Электронный ресурс]. – URL : https://insights.stackoverflow.com/survey/2020 (дата обращения : 02.11.2021).
11. Шолле, Ф. Глубокое обучение на Python [Электронный ресурс]. / Ф. Шолле. – СПб. : Питер, 2018. – 400 с. : ил.
12. Keras: библиотека глубокого обучение на Python [Электронный ресурс] // Русскоязычная документация Keras. – URL : https://ru-keras.com/home/ (дата обращения : 17.10.2021).