

УДК 378

С. Д. Коткин

S. D. Kotkin

Коткин Сергей Дмитриевич, к.п.н., доцент кафедры ИОТД, Новокузнецкий филиал (институт) Кемеровского государственного университета, г. Новокузнецк, Россия.

Kotkin Sergey Dmitrievich, Candidate of Pedagogical Sciences, Associate Professor, Novokuznetsk branch (institute) of Kemerovo State University, Novokuznetsk, Russia.

ОСОБЕННОСТИ ПРЕПОДАВАНИЯ КУРСА «ПРОГРАММИРОВАНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ» БУДУЩИМ УЧИТЕЛЯМ ТЕХНОЛОГИИ

FEATURES OF TEACHING THE COURSE «PROGRAMMING INTELLECTUAL SYSTEMS» TO FUTURE TECHNOLOGY TEACHERS

Аннотация. В статье рассматривается влияние Концепции преподавания учебного предмета «Технология» на содержание подготовки учителей данной предметной области. Описывается опыт изучения нейронных сетей с использованием таких инструментальных средств как язык программирования Python и библиотека Keras.

Abstract. The article examines the influence of the Concept of teaching the educational subject «Technology» on the content of teacher training in this subject area. The experience of studying neural networks using such tools as the Python programming language and the Keras library is described.

Ключевые слова: концепция преподавания учебного предмета «Технология», технологическое образование, машинное обучение, нейронные сети.

Keywords: concept of teaching the subject «Technology», technology education, machine learning, neural networks.

Проблема, которая всегда занимала педагогов, являлась реакция в изменении содержания образования в связи с изменениями социального и технологического укладов. Установившийся на протяжении последних пяти веков общественно-экономический уклад, характеризуется имманентными тратами на инновации, что привело к взрывному развитию технологий [1] и, как следствие, к резкому увеличению сложности структуры современных знаний. Последнее увеличило как потребность в квалифицированных трудовых кадрах, так и временные затраты на их подготовку. Однако, поскольку увеличение времени, затрачиваемого на обучение, уменьшает продолжительность трудовой деятельности, более желательным является подход, связанный с адаптацией профессионального образования, увеличения его вариативности при сохранении продолжительности сроков обучения. И если в предыдущем веке процессы специализации образования затрагивали преимущественно высшую школу, то сейчас, в частности, путем введения профильного обучения в старших классах, этот процесс переместился в среднее звено, что вылилось, в частности, в принятие в 2018 году Концепции преподавания учебного предмета «Технология» [2] (далее «Концепция»). Если ранее преобразовательная деятельность, изучаемая в предметной области «Технология» во многом была позаимствована из трудового обучения, то данная концепция делает упор на необходимость знакомства в ходе технологического обучения, с современными, в том числе, информационными технологиями, что необходимо, в том числе, для профессиональной ориентации и самоопределения личности. Из перечисленных в документе направлений, значительная часть, а именно компьютерное черчение, 3D-моделирование, технологии цифрового производства, аддитивные технологии, робототехника и системы автоматического управления, технологии умного дома и интернета вещей непосредственно связаны с информационными технологиями. Помимо модернизации технологического образования в школе в концепции упоминается и необходимость освоения предметной области «технология» в системе дополнительного образования. Всё это, по мнению авторов Концепции, нуждается в кадровом обеспечении, что реализуется путем разработки и реализации образовательных программ, в том числе и бакалавриата, по направлению «Педагогическое образование».

В рамках реализации данной концепции в Новокузнецком институт (филиале) Кемеровского государственного университета в подготовке бакалавров по специальности 44.03.05 «Педагогическое образование (с двумя профилями)» профиля «Технология и Дополнительное образование» была включена дисциплина «Программирование интеллектуальных систем». Задачей данной дисциплины является дать представление о современном состоянии систем искусственного интеллекта и областях их применения в приложении к означенным в Концепции направлениям. При выборе содержания обучения стояла задача точного определения целей обучения и выбора исходя из них содержания дисциплины. Уточненный выбор цели был обусловлен как содержанием Концепции, так и содержанием предыдущей подготовки. Применительно к приведенным в концепции направлениям могут использоваться информационные системы различных классов, применяющие различные по своей природе алгоритмы. Так, задачи, возникающие в компьютерном черчении и 3D-моделировании, хорошо решаются с помощью детерминированных алгоритмов, требующих для своей реализации хорошей математической подготовки, что делает рациональным изучение принципов построения подобных систем в подготовке учителей математики со вторым профилем «Информатика». Однако робототехника, системы автоматического управления, технологии умного дома и интернета вещей создаются и с применением систем искусственного интеллекта и машинного обучения. Традиционные системы искусственного интеллекта, представляющие собой системы, основанные на правилах, требуют создание абстрактных представлений конкретных ситуаций, что зачастую выполняется человеком. Поэтому такие системы чаще всего используются для решения некоторых абстрактных задач, например, вывода и доказательства теорем, расчета ходов в логические игры, экспертные системы, рассчитанные на использование специалистом, способным сформулировать абстрактное описание проблемной ситуации [3, 4]. Также к системам искусственного интеллекта принято относить системы машинного обучения, при этом в традиционных системах пространство признаков, с помощью которого описывается проблемная ситуация, проектируется вручную. Более современным направлением систем искусственного интеллекта являются системы обучения представлений, которые не только учатся на основе описания задачи в пространстве признаков находят её решение, но и создавать представление этой задачи без участия человека [5]. К этому классу систем относятся и системы глубокого обучения, построенные на искусственных нейронных сетях, чье теоретическое развитие началось ещё в пятидесятых годах двадцатого века, однако практическое применение стало возможным благодаря росту производительности вычислительной техники и появлению больших наборов данных, пригодных для создания обучающих выборок, лишь в последнее десятилетие. Системы глубокого обучения способны решать задачи самых разнообразных классов. Это как задачи, которые ранее решались другими методами, такие как задачи классификации, регрессии, кластеризации, снижения размерности, подавления шума, так и задачи, для которых другие методы решения не позволяют получать удовлетворительные результаты только в частных случаях, например, распознавание образов, машинный перевод, генерация текста, изображений, голоса или музыки, а также вычислительные задачи, которые из-за высокой вычислительной

сложности сейчас невозможно или неприемлемо долго решать с помощью детерминированных алгоритмов, например, задачи фолдина белков [6], расчет перемещения роботов-манипуляторов [7], расчет ходов в игре Го и т.п.

При создании нейронной сети основную роль играют обучающие данные – тот набор данных, на котором сеть будет обучаться, и её архитектура – то, из какого типа искусственных нейронов она будет построена и как они будут в ней соединены. Для создания обучающего набора данных используются различные источники данных, как коммерческие, так и открытые, которые специальным образом подготавливаются, что, как правило, требует участия человека. А так как качество нейронных сетей зависит от объема этого набора, то его создание зачастую является крайне дорогим этапом. Начальная архитектура нейронной сети выбирается исходя из имеющегося опыта её разработчиков. В дальнейшем, в зависимости от результатов обучения нейронной сети, архитектура может кардинальным образом меняться или настраиваться для достижения лучшей точности. При этом настройка архитектуры и параметров нейронной сети может быть автоматизирована. Самым требовательным к вычислительным ресурсам является этап обучения нейронных сетей, когда для подбора весовых коэффициентов искусственных нейронов обучающие данные многократно обрабатываются нейронной сетью, и при каждом проходе в зависимости от ошибки нейронной сети методом обратного распространения рассчитываются новые значения коэффициентов. Одной из самых больших проблем при создании нейронных сетей может быть непригодность конкретной архитектуры или конкретных параметров для определенной задачи, из-за чего процесс обучения приходится многократно повторять, изменяя нейронную сеть и не имея при этом никаких гарантий относительно полученного результата. Однако в случае типовых задач возможные виды архитектуры достаточно хорошо изучены и дают предсказуемые результаты.

При обучении созданию и применению нейронных сетей студентов направления 44.03.05 «Педагогическое образование (с двумя профилями)» профиля «Технология и Дополнительное образование» мы исходили из того, что они в своей деятельности, как правило, будут применять типовые архитектуры нейронных сетей в задачах технического творчества, что позволило нам сократить объем изучаемых инструментальных средств, исключив, в частности, средства для автоматизации подбора параметров и гиперпараметров нейронной сети, средства для определения влияния обучающих данных на различные слои нейронов и т.д. При этом в качестве базового инструментального средства нами был выбран язык программирования Python, используемый совместно с библиотекой Keras, использующей для выполнения операций над тензорами библиотеку TensorFlow [8]. Такое сочетание имеет несколько преимуществ. Во-первых, библиотека TensorFlow в настоящее время активно развивается и поддерживает большинство современного оборудования. В частности, для ускорения вычислений при обучении она способна использовать ресурсы графического процессора компьютера. Однако при этом она имеет весьма сложный интерфейс для взаимодействия с ней из языков программирования. Для упрощения этого интерфейса, в том числе, для сокрытия значительной части рутинных операций по созданию нейронной сети, используется библиотека Keras. Эта библиотека создает дополнительный абстрактный слой между одной из библиотек низкого уровня – помимо TensorFlow Keras может использовать также CNTK или Theano [9]. При этом общее управление процессом обучения нейронной сети осуществляется с помощью языка программирования Python. Это интерпретируемый язык, позволяющий как выполнять команды в интерактивном режиме, так и в составе программ. Он имеет встроенную поддержку многомерных списков, что упрощает подготовку данных для нейронных сетей. Кроме того, его можно использовать совместно с инструментальным средством Jupiter Notebook, который позволяет комбинировать части программ, интерактивные команды, результаты выполнения команд, в том числе, содержащие графику, а также текстовые пояснения.

Используя такое сочетание инструментальных средств мы, помимо изучения теоретических сведений о нейронных сетях, учимся создавать нейронные сети, которые могут быть применены в создании систем автоматического управления, использования технологии умного дома и интернета вещей.

Список литературы

1. Хазин, М. Л. Воспоминания о будущем. Идеи современной экономики [Текст]. / М. Л. Хазин. – М. : Группа Компаний «РИПОЛ классик», «Сфера», 2019. – 436 с.: ил.
2. Концепция преподавания учебного предмета «Технология» [Электронный ресурс]. // Банк документов. Министерство просвещения Российской Федерации, 2018. – URL : <https://docs.edu.gov.ru/id501> (дата обращения : 25.09.2020).
3. Осуга, С. Обработка знаний Текст]. / С. Осуга, пер. с япон. [– М. : Мир, 1989. – 293 с., ил.

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-promyshlennyyh-robotov-manipulyatorov-na-osnove-neyronnyh-setey> (дата обращения : 28.11.2020).
8. Шолле, Ф. Глубокое обучение на Python [Текст]. / Ф. Шолле. – СПб. : Питер, 2018. – 400 с.: ил.
9. Keras : библиотека глубокого обучение на Python [Электронный ресурс]. // Русскоязычная документация Keras. – URL : <https://ru-keras.com/home/> (дата обращения : 17.11.2020).