

УДК 661.63
DOI: 10.15827/0236-235X.135.390-398

Дата подачи статьи: 22.05.21
2021. Т. 34. № 3. С. 390–398

Программа формирования стека методов искусственного интеллекта при решении прикладных задач

А.Ю. Пучков¹, к.т.н., доцент, putchkov63@mail.ru

Е.И. Лобанева¹, аспирант, lobaneva94@mail.ru

М.А. Василькова¹, студент, vasilkova_mariya00@mail.ru

¹ Смоленский филиал Национального исследовательского университета МЭИ,
г. Смоленск, 214013, Россия

Предложен алгоритм автоматизированного выбора стека методов искусственного интеллекта для обработки данных на основе учета их характеристик, таких как объем и формат представления, а также специфики задачи, решаемой на основе этих данных.

Цель исследования – разработка системы оценок и критериев для выбора стека методов искусственного интеллекта. Это позволит наиболее эффективно реализовать требуемые целевые параметры обработки данных.

Задача исследования заключалась в анализе проблемы выбора парадигмы методов искусственного интеллекта, в определении системы оценок, наиболее полно отражающих особенности предметной области, для которой выбирается парадигма, и в разработке программы, формирующей стек методов обработки данных. Учитывая многообразие методов искусственного интеллекта, для конкретизации задачи исследования рассматривались методы машинного обучения. Для обеспечения возможности применения более простых методов анализа для получаемых данных был предусмотрен выбор статистических методов, также относящихся к технологии машинного обучения.

Разработанный алгоритм выбора стека методов искусственного интеллекта содержит многоступенчатую процедуру. Она включает в себя следующее: оценка важности параметров с точки зрения их влияния на целевой параметр; применение системы нечеткого вывода, которая на основе задаваемых экспертных оценок характеристик данных ранжирует предложенный набор методов обработки; подбор сайтов и поиск в них ключевых слов предметной области, отражающих задачу, для решения которой применяются обрабатываемые данные.

Представлены результаты работы программы, написанной на языке MatLAB и реализующей предложенный алгоритм, которые демонстрируют корректный выбор методов искусственного интеллекта на основе заданных характеристик входной информации.

Ключевые слова: выбор методов искусственного интеллекта, машинное обучение, оценка прагматической ценности информации.

Переход на технологический уклад четвертой промышленной революции (Индустрия 4.0) подразумевает массовое внедрение цифровых решений на всех этапах жизненного цикла сложных киберфизических систем, характеризующихся сложностью протекающих в них взаимосвязанных процессов, многомасштабностью применяемых моделей, разной структурированностью стоящих перед ними задач. Предполагается, что трансформация производственных процессов будет идти в направлении внедрения искусственного интеллекта, роботов, имплантированных технологий, взаимодействующих и координирующих машин, систем автономного решения проблем, 3D-печати и других [1–3].

Отмеченным трансформациям способствует активное внедрение в производственные процессы интернета вещей (Internet of

Things, IoT) и пятого поколения мобильной связи, действующей на основе стандартов телекоммуникаций 5G/IMT-2020. Такие сети значительно повышают пропускную способность по сравнению с технологиями 4G и делают возможным надежный обмен данными между устройствами (device-to-device). Это приводит к лавинообразному росту объемов технологической информации, только 5 % которой используется организациями для повышения операционной эффективности, что, безусловно, инспирирует производственный менеджмент совершенствовать информационное обеспечение технологических процессов для поддержания конкурентного преимущества [4].

При реализации такого совершенствования необходимо учитывать иерархическую структуру информационных систем на крупных предприятиях, обеспечивающую стратифика-

цию решаемых задач, начиная от самого нижнего уровня обработки контрольно-измерительной информации и управления отдельными участками технологического процесса и заканчивая рекомендательными системами на уровне высшего менеджмента. Однако такая структура организации информационного пространства предприятия требует от разработчиков информационных систем на каждом уровне иерархии осуществлять формирование стека (набора) технологий, позволяющего наиболее эффективно решать стоящие на этих уровнях задачи. Применение в современных киберфизических системах промышленного интернета вещей (Industrial IoT) также способствует генерации огромных объемов данных, поэтому для их обработки и анализа создан стек технологий больших данных (Big Data), включающий широкий спектр интеллектуальных методов, в том числе машинного обучения (Machine Learning, ML) [5–7].

Отметим, что в стек ML входят и статистические методы, имеющие давнюю историю применения, поэтому их использование также стоит рассматривать как более простую альтернативу методам искусственного интеллекта, если это целесообразно исходя из поставленной задачи и характеристик исходных данных. Однако при выборе методов искусственного интеллекта ситуация меняется из-за значительно большего числа этих методов и их модификаций по сравнению со статистическими. Особенно это касается моделей на основе глубоких искусственных нейронных сетей (Deep neural network, DNN), для которых созданы десятки архитектур и сотни предобученных сетей для решения конкретных практических задач.

Современные DNN содержат миллионы и миллиарды настраиваемых параметров. Так, появившаяся в середине 2020 года сеть GPT 3 (Generative Pre-trained Transformer 3) от компании OpenAI содержит 175 млрд настраиваемых параметров [8]. Обучение GPT 3, которое проводилось на суперкомпьютере Microsoft Azure AI, стоило более 4,6 млн долларов – столь большие затраты непосильны большинству компаний, однако обученная сеть доступна для использования (пока OpenAI разрешила закрытый API на базе GPT-3, доступ к которому получили только избранные разработчики). Для разработчика выбор из большого «зоопарка» предобученных сетей наиболее подходящих средств решения задач становится все труднее и требует высоких профессиональных

компетенций, а также времени, поэтому актуальна автоматизация этой процедуры. Решение данной исследовательской задачи позволит сократить время на подбор нужного алгоритмического инструментария для обработки и анализа технологической информации и снизить требования к уровню компетенции разработчика за счет использования банка готовых моделей обработки данных.

Понимание необходимости создания рекомендательных систем, позволяющих разработчику, не являющемуся специалистом в области архитектур ML, осуществлять их обоснованный выбор на практике, обусловило создание целого комплекса алгоритмов, автоматизирующих трудоемкую процедуру подбора моделей ML и их гиперпараметров [9–11]. Отметим, что под гиперпараметрами понимается набор характеристик алгоритмов ML, которые устанавливаются перед запуском процесса обучения, а не определяются в его процессе.

Лидеры рынка информационных технологий также создают свои инструменты для автоматизации поиска архитектур. Например, компания Google в 2019 году предоставила в открытый доступ фреймворк, автоматизирующий поиск архитектуры DNN распознавания разговорной речи и выявления ключевых слов. В его основе лежит настраиваемый генетический алгоритм, позволяющий создавать DNN с на порядок меньшим, чем у лучших актуальных моделей, количеством обучаемых параметров, при этом обеспечивается повышение производительности до 4,09 % при классификации фонем в процессе выделения ключевых слов в тексте [12]. Проводятся исследования и по автоматизации хранения моделей машинного обучения, например, в [13] предложена система хранения ансамблей нейросетевых моделей, обеспечивающая структурированное хранение данных на различных этапах решения задач прогнозирования временных рядов. Однако в большинстве случаев выбор архитектуры моделей ML разработчик выполняет сам, опираясь на собственный опыт и готовые решения, представленные на различных веб-ресурсах, например, Kaggle – международная платформа для конкурсов по обработке данных и машинному обучению при решении серьезных и актуальных задач, AI Russia – российский конкурс и библиотека проектов на основе искусственного интеллекта, Browse State-of-the-Art, Awesome Hand Pose Estimation и другие. Автоматизировать процесс изучения материалов сайтов сложно, так как нет единого

классификатора решений, позволяющего по ряду дескрипторов заполнять единый репозиторий и уже из него выбирать нужное решение, но упростить хотя бы начальные этапы этого процесса вполне возможно.

Процесс поиска подходящей модели ML можно представить как многоэтапную процедуру. На *первом этапе* в данных выбираются целевые параметры и осуществляется анализ важности остальных параметров с точки зрения их влияния на целевые. Для этого можно использовать статистические методы корреляционного анализа или алгоритм «случайный лес» [14]. Затем часть параметров исключаются из дальнейшего рассмотрения, что позволяет уменьшить размерность пространства признаков. На *втором этапе* определяется набор оценок и критериев, учитывающих особенности предметной области, формы представления данных, которые используются на следующем этапе. На *третьем этапе* выбирается парадигма реализации ML-модели. На *заключительном этапе* осуществляется конкретизация ML-модели на основании опыта разработчика и анализа информации на веб-ресурсах в рамках выбранной парадигмы.

Целью проведенного исследования являлась разработка критериев и системы оценок имеющейся информации о предметной области для обоснованного выбора парадигмы интеллектуальной обработки данных, а также алгоритма, позволяющего реализовать требуемые целевые параметры этой обработки.

Задача исследования состояла в проведении анализа существующего состояния проблемы выбора парадигмы интеллектуальной обработки данных и определении системы оценок, наиболее полно отражающих особенности предметной области, для которой выбирается парадигма. Также в задачу входило создание ПО, реализующего алгоритм выбора парадигмы методов искусственного интеллекта для обработки данных.

Научную новизну результатов представленных исследований составляют предложенная система оценок и критериев, применяемая при выборе парадигмы ML-модели обработки данных, и алгоритм, реализующий этот выбор. Практическая значимость работы заключается в созданном ПО, использующем указанную систему оценок и критериев и применяемом в алгоритмическом обеспечении рекомендательных систем.

Предлагаемая система оценок базируется на принятых характеристиках больших данных,

обозначаемых в англоязычной литературе как 5V – Volume, Variety, Velocity, Veracity, Value (объем, разнородность, скорость, достоверность, ценность) [15]. Иногда к этим характеристикам добавляют Variability и Visualization (изменчивость и визуализация): изменчивость предполагает, что данные могут иметь разный смысл в разном контексте, а визуализация акцентирует внимание на том, что для человеческого восприятия более наглядны именно визуальные образы. Однако на этапе формирования стека интеллектуальных методов эти характеристики не являются критичными, поэтому были исключены из предлагаемой системы оценок. В представленном списке под достоверностью подразумевается отбор проверенных (с отсеиванием информационного «шума») данных, а под ценностью – их ожидаемая полезность по какому-либо критерию, например, прагматическая ценность информации, выражаемая как экономические эффекты от применения результатов обработки.

Разнообразие характеристик приводит к сложности их унифицированного количественного описания, которое позволило бы свести к общему категориальному аппарату процедуру обеспечения требуемых целевых параметров обработки, и вполне обоснованы применение методов нечеткой логики и описание наборов данных в терминах нечетких лингвистических переменных и с использованием нечетких продукционных правил. Нечетко логические подходы хорошо зарекомендовали себя в задачах выбора альтернатив с нечетко заданными значениями критериев выбора [16]. Для применения в задаче выбора технологии искусственного интеллекта указанные характеристики слишком обобщены и приводят к большой вариативности реализации методов обработки данных, поэтому конкретизируем часть этих характеристик.

Разнородность данных может рассматриваться как многообразие их структур и форматов представления (числовая, текстовая, графическая, аудиоданные), согласованности временных параметров поступления информации, нечеткости, неполноты и других. При обработке многими интеллектуальными методами наиболее существенным является разделение данных на изображения и последовательности, так как, в конечном счете, текстовая, звуковая и числовая информация является последовательностью чисел, обрабатываемых однотипными методами, в то время как изображения требуют других подходов при выявлении высо-

коуровневых абстракций в данных. Поэтому Variety в дальнейшем будем рассматривать с позиции соотношения данных, представляемых как последовательности и изображения, а также наличия в данных нечеткости и плохой структурированности.

Достоверность данных, в понимании устранения «шума», обеспечивается за счет их предобработки или компенсации шумовых составляющих алгоритмическими особенностями интеллектуальных методов, поэтому при реализации алгоритма выбора стека технологий эта характеристика не будет обрабатываться отдельно.

Характеристика скорости также не будет учитываться, так как сложные ML-модели предполагают длительный период обучения на больших наборах данных (до нескольких дней и недель), но после обучения они срабатывают достаточно быстро. Это обеспечивает работу даже в режиме реального времени, например, в автопилотах на транспорте [17, 18]. Поэтому скорость поступления информации при выборе стека технологий не будет рассматриваться как конкретизирующий признак в предположении, что он учитывается на этапе подготовки датасетов и в обучении.

В определение характеристики ценности данных может вкладываться интегральный смысл, количественно отражаемый обобщенным критерием, учитывающим как экономические эффекты от результатов использования получаемых из Big Data глубинных закономерностей, так и эффекты, достигаемые за счет экономии аппаратно-вычислительных ресурсов. Последние возникают за счет исключения из обработки маловажных параметров, не приносящих значительный прирост точности результатов при обработке данных. Как уже отмечалось, оценка важности параметров происходит на первом этапе процедуры поиска модели ML, поэтому характеристика Value не рассматривается при выборе стека технологий искусственного интеллекта. Конкретный вид интеллектуальной задачи как отдельная характеристика не учитывался, так как в разнообразных задачах ML они в большинстве своем сводятся к задачам регрессии и классификации (или кластеризации).

В результате система оценок, на основе которой проводится выбор стека технологий, можно представить как кортеж нечетких переменных: $D = \langle x_1 = Volume, Vl, x_2 = Sequence/picture, SP, x_3 = Fuzziness, Fz \rangle$, в котором каждую переменную представим набором термов

(например, «малое», «среднее», «большое» и др.), нормированных к диапазону $[0; 1]$. Оценка *Sequence/picture* введена для более точной дифференцировки парадигмы DNN, отражающей соотношение данных, представленных в форме временных рядов (для их обработки предпочтительно применение рекуррентных нейронных сетей – Recurrent Neural Network, RNN) и изображений (в том числе видеоданных), для анализа которых применяются сверточные нейронные сети (Convolutional Neural Network, CNN) [19].

Критерий выбора стека технологий C формируется как база нечетких продукционных правил, наполняемая экспертами предметной области и отражающая их опыт разработки, а также результаты применения различных интеллектуальных методов другими исследователями:

$$\left\{ \begin{array}{l} \bigcup_{i=1}^n \text{if } x_1 = F_{j_1, j_1 \in \{1,2,\dots\}} \wedge x_2 = \\ = F_{j_2, j_2 \in \{1,2,\dots\}} \wedge \dots \wedge x_5 = F_{j_5, j_5 \in \{1,2,\dots\}}, \text{ then } C_1 = FC_1, \\ \bigcup_{i=1}^n \text{if } x_1 = F_{j_1, j_1 \in \{1,2,\dots\}} \wedge x_2 = \\ = F_{j_2, j_2 \in \{1,2,\dots\}} \wedge \dots \wedge x_5 = F_{j_5, j_5 \in \{1,2,\dots\}}, \text{ then } C_2 = FC_2, \\ \dots \\ \bigcup_{i=1}^n \text{if } x_1 = F_{j_1, j_1 \in \{1,2,\dots\}} \wedge x_2 = \\ = F_{j_2, j_2 \in \{1,2,\dots\}} \wedge \dots \wedge x_5 = F_{j_5, j_5 \in \{1,2,\dots\}}, \text{ then } C_m = FC_m, \end{array} \right.$$

где F_j – терм-множество нечетких переменных из D ; FC_m – терм-множество значений критерия, отражающего целесообразность применения технологий под номером nt для соответствующей комбинации значений нечетких переменных из кортежа D .

В результате срабатывания нечеткой системы получаем вектор четких значений $C^* = [C_1^*, C_2^*, \dots, C_m^*]$. Ранжирование элементов C^* по убыванию позволяет расположить технологии по степени целесообразности их применения для обработки входных данных с указанными характеристиками.

Для апробации предложенного метода выбора технологий искусственного интеллекта была разработана программа в среде MatLAB 2021a, предоставляющей инструментарий fuzzy logic toolbox для создания нечетких систем. На рисунке 1 показана структура примененной системы нечеткого вывода (использовался алгоритм Мамдани) для следующего укрупненного стека технологий искусственного интеллекта обработки данных: $C = \{C_1 = \text{«деревья решений и статистические методы»}, C_2 = \text{«системы нечеткого вывода»}, C_3 = \text{«неглу-»}$

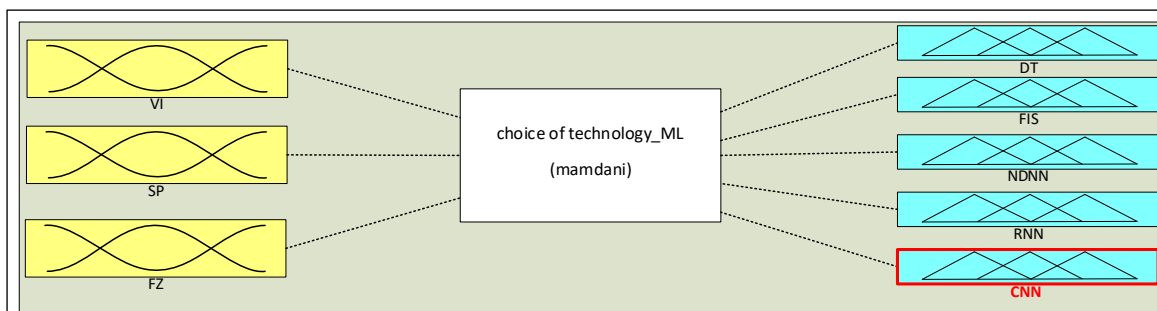


Рис. 1. Структура системы нечеткого вывода

Fig. 1. A fuzzy inference system structure

бокые нейронные сети», $C_4 = \text{«RNN»}$, $C_5 = \text{«CNN»}$. Здесь в стек «системы нечеткого вывода» отнесены также такие ML-модели, как адаптивные сети на основе системы нечеткого вывода (adaptive neuro-fuzzy inference system, ANFIS). В стек «деревья решений и статистические методы» входят алгоритмы линейного машинного обучения (логистическая регрессия, линейный дискриминантный анализ) и нелинейные алгоритмы (к-ближайших соседей, наивный байесовский, деревья классификации и регрессии, метод опорных векторов).

При программной реализации наименования технологий стека были заменены на латинские названия, так как кириллические имена переменных в MatLAB не поддерживаются: $C = \{C_1 = \text{«decision trees and statistical methods, DT»}, C_2 = \text{«fuzzy inference systems, FIS»}, C_3 = \text{«not deep neural networks, NDNN»}, C_4 = \text{«RNN»}, C_5 = \text{«CNN»}\}$.

Наполнение базы знаний (представляющей в данном случае множество нечетких продукционных правил) требует значительных временных затрат. Наличие трех термов для каждой входной нечеткой переменной («малое», «среднее» и «большое» значения) при трех вхо-

дах приводит к необходимости задания $3^3 = 27$ правил, если обеспечивать в правилах перебор всех возможных комбинаций (без учета правил, использующих значения none для нечетких переменных). Фрагмент базы правил, иллюстрирующий состав переменных, участвующих в ее формировании, представлен на рисунке 2.

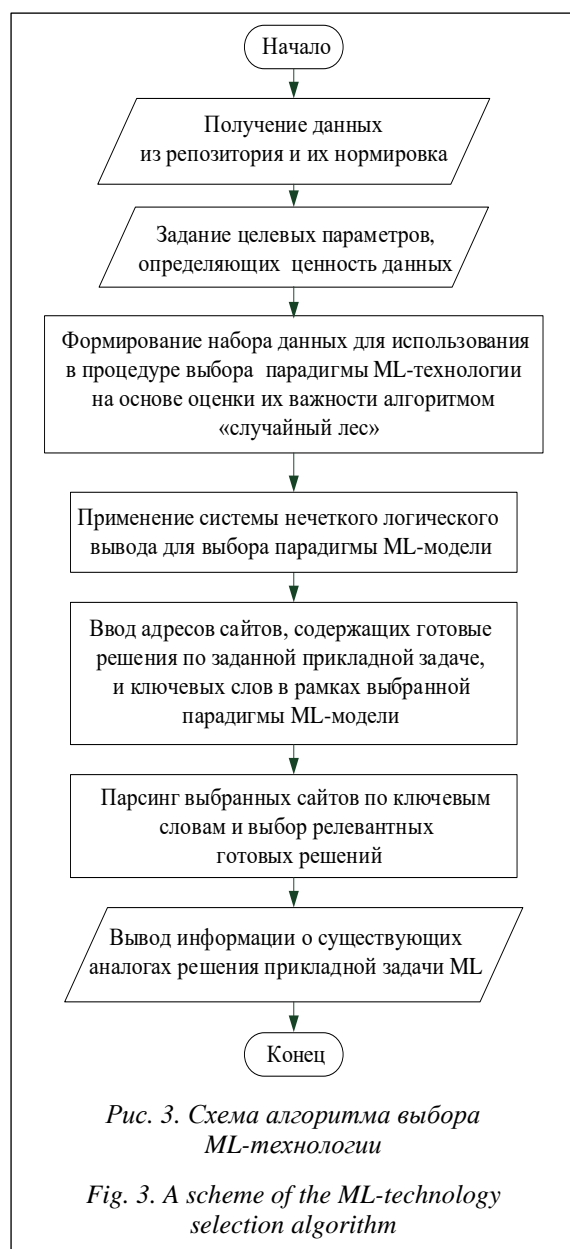
Укрупненная схема алгоритма работы ПО, реализующего этапы поиска модели ML, показана на рисунке 3. При оценке важности параметров с помощью алгоритма «случайный лес» использовалась функция templateTree, обучался ансамбль из 200 деревьев регрессии (функция fitrensemble).

Для разработки графического интерфейса программы был использован инструмент App Designer среды MatLAB (см. пример главной формы программы <http://www.swsys.ru/uploaded/image/2021-3/2021-3-dop/5.jpg>). В начале работы с программой надо указать характеристики данных с помощью перемещения ползунков и нажать кнопку «Calculate technology stack parameters», в результате произойдет срабатывание системы нечеткого вывода и в разделе «Technology preference» отразятся значе-

1. If (VI is mf1) and (SP is mf1) and (Fz is mf1) then (DT is mf3)(FIS is mf1)(NDNN is mf3)(RNN is mf1)(CNN is mf1) (1)
2. If (VI is mf1) and (SP is mf1) and (Fz is mf2) then (DT is mf2)(FIS is mf2)(NDNN is mf2)(RNN is mf1)(CNN is mf1) (1)
3. If (VI is mf1) and (SP is mf1) and (Fz is mf3) then (DT is mf3)(FIS is mf3)(NDNN is mf1)(RNN is mf1)(CNN is mf1) (1)
4. If (VI is mf2) and (SP is mf1) and (Fz is mf1) then (DT is mf2)(FIS is mf1)(NDNN is mf2)(RNN is mf2)(CNN is mf1) (1)
5. If (VI is mf2) and (SP is mf2) and (Fz is mf1) then (DT is mf2)(FIS is mf2)(NDNN is mf2)(RNN is mf2)(CNN is mf2) (1)
6. If (VI is mf2) and (SP is mf3) and (Fz is mf2) then (DT is mf1)(FIS is mf1)(NDNN is mf2)(RNN is mf1)(CNN is mf3) (1)
7. If (VI is mf2) and (SP is mf3) and (Fz is mf3) then (DT is mf2)(FIS is mf2)(NDNN is mf1)(RNN is mf1)(CNN is mf2) (1)
8. If (VI is mf3) and (SP is mf1) and (Fz is mf1) then (DT is mf1)(FIS is mf1)(NDNN is mf2)(RNN is mf3)(CNN is mf2) (1)
9. If (VI is mf3) and (SP is mf2) and (Fz is mf1) then (DT is mf1)(FIS is mf1)(NDNN is mf2)(RNN is mf2)(CNN is mf2) (1)
10. If (VI is mf3) and (SP is mf3) and (Fz is mf1) then (DT is mf1)(FIS is mf1)(NDNN is mf1)(RNN is mf1)(CNN is mf3) (1)
11. If (VI is mf3) and (SP is mf3) and (Fz is mf2) then (DT is mf2)(FIS is mf2)(NDNN is mf1)(RNN is mf2)(CNN is mf2) (1)
12. If (VI is mf3) and (SP is mf3) and (Fz is mf3) then (DT is mf1)(FIS is mf1)(NDNN is mf1)(RNN is mf1)(CNN is mf3) (1)

Рис. 2. База нечетких продукционных правил

Fig. 2. A fuzzy production rule base



ния функций принадлежности – чем больше значение, тем предпочтительнее использование данной ML-технологии.

Далее пользователь сам анализирует полученные значения функций принадлежности и в разделе «Selection of thematic sites» указывает номер технологии, по которой он хотел бы провести подбор материалов, присутствующих на web-ресурсах (их адреса автоматически появляются после указания номера технологии). Необходимо выделить строку с адресом интересующего сайта, задать список ключевых слов и нажать кнопку «web scraping».

В результате этих действий в поле web scraping results отразятся найденные на заданной web-странице ключевые слова и количество их вхождений, что может быть использовано разработчиком для более детального ознакомления с материалом сайта.

Программа парсинга сайтов выполнена с использованием встроенных функций MATLAB, таких как webread, htmlTree, extractHTMLText, однако в результате ее работы пока лишь указывается количество найденных ключевых слов. В дальнейшем предполагается разработка более таргетного поиска ресурсов, возможно, с подключением специализированных сторонних программ для парсинга.

Проведенное исследование позволило сформировать систему оценок и критериев на основе методов нечеткой логики, которые использованы в разработанном алгоритме, автоматизирующем процедуру выбора стека методов искусственного интеллекта, применяемых для обработки данных в прикладных задачах. Указанный алгоритм реализован, и программа может найти применение в рекомендательных системах как составной элемент их информационного обеспечения.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-37-90062 Аспиранты.

Литература

1. Oztemel E., Gursev S. Literature review of Industry 4.0 and related technologies. Journal of Intelligent Manufacturing, 2020, vol. 31, no. 1, pp. 127–182. DOI: 10.1007/s10845-018-1433-8.
2. Черновалова М.В. Нечеткие прецедентные модели для управления проектами с использованием мультионтологического подхода // Прикладная информатика. 2021. Т. 16. № 2. С. 4–16. DOI: 10.37791/2687-0649-2021-16-2-4-16.
3. Дли М.И., Булыгина О.В., Соколов А.М. Рубрицирование текстовой информации на основе голосования интеллектуальных классификаторов // Прикладная информатика. 2020. Т. 15. № 5. С. 29–36. DOI: 10.37791/2687-0649-2020-15-5-29-36.
4. Соколов Д.И., Соловьев С.Ю. Контроль и мониторинг промышленного оборудования с использованием платформы MindSphere компании Siemens. ИСУП. 2018. № 4. С. 57–62.

5. Groppe S. Emergent models, frameworks, and hardware technologies for Big data analytics. The Journal of Supercomputing, 2020, vol. 76, no. 3, pp. 1800–1827. DOI: 10.1007/s11227-018-2277-x.
6. Кулыгин О.П. Применение больших данных (big data) в проектировании экспертных систем // Прикладная информатика. 2020. Т. 15. № 6. С. 130–141. DOI: 10.37791/2687-0649-2020-15-6-130-141.
7. Дли М.И., Власова Е.А., Моргунова Э.В., Соколов А.М. Создание цифрового двойника химико-технологической системы с использованием языка Python // Прикладная информатика. 2021. Т. 16. № 1. С. 22–31. DOI: 10.37791/2687-0649-2021-16-1-22-31.
8. Brown T.B., Mann B., Ryder N. et al. Language Models are Few-Shot Learners. ArXiv.org. URL: <https://arxiv.org/abs/2005.14165> (дата обращения: 20.05.2021).
9. Luo G. A review of automatic selection methods for machine learning algorithms and hyper-parameter values. Netw Model Anal Health Inform Bioinforma, 2016, vol. 5, pp. 1–15. DOI: 10.1007/s13721-016-0125-6.
10. Пучков А.Ю., Дли М.И. Алгоритм настройки гиперпараметров сверточной нейронной сети в задаче классификации объектов // ММТТ. 2018. Т. 4. С. 47–50.
11. Дли М.И., Пучков А.Ю., Лобанева Е.И. Анализ влияния архитектуры входных слоев свертки и подвыборки глубокой нейронной сети на качество распознавания изображений // Прикладная информатика. 2020. Т. 15. № 1. С. 113–122.
12. Mazzawi H., Gonzalvo X., Kracun A., Sridhar P., Subrahmanya N.A. et al. Improving keyword spotting and language identification via neural architecture search at scale. Interspeech, 2019, pp. 1278–1282. DOI: 10.21437/Interspeech.2019-1916.
13. Пучков Е.В. Разработка системы хранения ансамблей нейросетевых моделей // Программные продукты и системы. 2017. Т. 23. № 1. С. 12–20. DOI: 10.15827/0236-235X.117.012-020.
14. Liu Y., Zhao H. Variable importance-weighted Random Forests. Quantitative Biology, 2017, vol. 5, no. 4, pp. 338–351. DOI: 10.1007/s40484-017-0121-6.
15. Ngai E.W.T., Gunasekaran A., Wamba S.F., Akter Sh., Dubey R. Big data analytics in electronic markets. Electron Markets, 2017, vol. 27, pp. 243–245. DOI: 10.1007/s12525-017-0261-6.
16. Пучков А.Ю. Программа модифицированного метода выбора недоминируемых альтернатив на основе нечеткого отношения предпочтения // Программные продукты, системы и алгоритмы. 2017. № 4. URL: <http://swsys-web.ru/program-of-a-modified-method-for-selecting-non-dominant-alternatives.html> (дата обращения: 20.05.2021).
17. Haq E.U., Huarong X., Xuhui C., Wanqing Z., Jianping F., Abid F. A fast hybrid computer vision technique for real-time embedded bus passenger flow calculation through camera. Multimedia Tools and Applications, 2020, vol. 79, pp. 1007–1036. DOI: 10.1007/s11042-019-08167-y.
18. Karkera T., Singh C. Autonomous bot using machine learning and computer vision. SN Computer Science, 2021, vol. 2, no. 4, art. 251. DOI: 10.1007/s42979-021-00640-6.
19. Дли М.И., Пучков А.Ю., Лобанева Е.И. Алгоритмы формирования изображений состояний объектов для их анализа глубокими нейронными сетями // Прикладная информатика. 2019. Т. 14. № 2. С. 43–55.

The program for forming a stack of artificial intelligence methods to solve applied problems

*A.Yu. Puchkov*¹, Ph.D. (Engineering), Associate Professor, putchkov63@mail.ru
*E.I. Lobaneva*¹, Postgraduate Student, lobaneva94@mail.ru
*M.A. Vasilkova*¹, Student, vasilkova_mariya00@mail.ru

¹ Smolensk Branch of the Moscow Power Engineering Institute, Smolensk, 214013, Russian Federation

Abstract. The paper proposes an algorithm for automated selection of a stack of artificial intelligence methods for data processing based on taking into account their characteristics, such as volume, presentation format, as well as the specifics of the problem solved based on these data.

The purpose of the study is to develop a system of assessments and criteria to select a stack (a list of methods reflecting different levels of their suitability for solving an applied problem) of artificial intelligence methods,

which makes it possible to effectively implement the required target parameters of data processing, as well as creating an algorithm to form such stack.

The task of the study was to analyze the problem of choosing a paradigm of artificial intelligence methods, to determine an assessment system that most fully reflects the features of the subject area for which the paradigm is selected, to develop a program that forms a stack of data processing methods. Considering the variety of artificial intelligence methods, and with the aim of concretizing the research task, machine learning methods were considered. To provide the possibility of applying simpler analysis methods to the obtained data, a choice of statistical methods also related to machine learning technology was provided.

The developed algorithm for choosing a stack of artificial intelligence methods contains a multi-stage procedure. It includes the following: assessment of the parameter importance in terms of their impact on the target parameter; the use of a fuzzy inference system that ranks the proposed set of processing methods based on the specified expert assessments of data characteristics; selection of websites and search in them for subject area keywords reflecting the task for the solution of which the processed data is used.

The paper presents the program work results written in the MatLAB language and implementing the proposed algorithm, which demonstrate the correct choice of artificial intelligence methods based on the given characteristics of the input information.

Keywords: choice of artificial intelligence methods, machine learning, assessment of the information pragmatic value.

Acknowledgements. The study was financially supported by the Russian Foundation for Basic Research within the framework of scientific project no. 20-37-90062 Postgraduates.

References

1. Oztemel E., Gursev S. Literature review of Industry 4.0 and related technologies. *Journal of Intelligent Manufacturing*, 2020, vol. 31, no. 1, pp. 127–182. DOI: 10.1007/s10845-018-1433-8.
2. Chernovalova M.V. Fuzzy case models for project management using a multi-ontology approach. *Journal of Applied Informatics*, 2021, vol. 16, no. 2, pp. 4–16. DOI: 10.37791/2687-0649-2021-16-2-4-16 (in Russ.).
3. Dli M., Bulygina O., Sokolov A. Rubrication of text information based on the voting of intellectual classifiers. *Journal of Applied Informatics*, 2020, vol. 15, no. 5, pp. 29–36. DOI: 10.37791/2687-0649-2020-15-5-29-36 (in Russ.).
4. Sokolov D.I., Solovev S.Yu. Industrial equipment control and monitoring using the Siemens MindSphere platform. *ISUP*, 2018, vol. 4, pp. 57–62.
5. Groppe S. Emergent models, frameworks, and hardware technologies for Big data analytics. *The Journal of Supercomputing*, 2020, vol. 76, no. 3, pp. 1800–1827. DOI: 10.1007/s11227-018-2277-x.
6. Kultygin O.P. Application of big data in the design of expert systems. *Journal of Applied Informatics*, 2020, vol. 15, no. 6, pp. 130–141. DOI: 10.37791/2687-0649-2020-15-6-130-141.
7. Dli M., Vlasova E., Sokolov A., Morgunova E. Creation of a chemical-technological system digital twin using the Python language. *Journal of Applied Informatics*, 2021, vol. 16, no. 1, pp. 22–31. DOI: 10.37791/2687-0649-2021-16-1-22-31.
8. Brown T.B., Mann B., Ryder N. et al. Language Models are Few-Shot Learners. *ArXiv.org*. Available at: <https://arxiv.org/abs/2005.14165> (accessed May 20, 2021).
9. Luo G. A review of automatic selection methods for machine learning algorithms and hyper-parameter values. *Netw Model Anal Health Inform Bioinforma*, 2016, vol. 5, pp. 1–15. DOI: 10.1007/s13721-016-0125-6.
10. Puchkov A.Yu., Dli M.I. Algorithm for tuning hyperparameters of a convolutional neural network in the object classification problem. *Mathematical Methods in Engineering and Technology*, 2018, vol. 4, pp. 47–50 (in Russ.).
11. Dli M.I., Puchkov A.Yu., Lobaneva E.I. Analysis of the influence of the architecture of the input layers of convolution and subsampling of a deep neural network on the quality of image recognition. *Journal of Applied Informatics*, 2020, vol. 1, pp. 113–122 (in Russ.).
12. Mazzawi H., Gonzalvo X., Kracun A., Sridhar P., Subrahmanya N.A. et al. Improving keyword spotting and language identification via neural architecture search at scale. *Interspeech*, 2019, pp. 1278–1282. DOI: 10.21437/Interspeech.2019-1916.
13. Puchkov E.V. Neural network ensembles storage development. *Software and Systems*, 2017, vol. 23, no. 1, pp. 12–20. DOI: 10.15827/0236-235X.117.012-020 (in Russ.).
14. Liu Y., Zhao H. Variable importance-weighted Random Forests. *Quantitative Biology*, 2017, vol. 5, no. 4, pp. 338–351. DOI: 10.1007/s40484-017-0121-6.

15. Ngai E.W.T., Gunasekaran A., Wamba S.F., Akter Sh., Dubey R. Big data analytics in electronic markets. *Electron Markets*, 2017, vol. 27, pp. 243–245. DOI: 10.1007/s12525-017-0261-6.
16. Puchkov A.Yu. The program of the modified method for choosing non-dominated alternatives based on a fuzzy preference relation. *Software Journal: Theory and Applications*, 2017, vol. 4. Available at: <http://swsys-web.ru/program-of-a-modified-method-for-selecting-non-dominant-alternatives.html> (accessed May 20, 2021) (in Russ.).
17. Haq E.U., Huarong X., Xuhui C., Wanqing Z., Jianping F., Abid F. A fast hybrid computer vision technique for real-time embedded bus passenger flow calculation through camera. *Multimedia Tools and Applications*, 2020, vol. 79, pp. 1007–1036. DOI: 10.1007/s11042-019-08167-y.
18. Karkera T., Singh C. Autonomous bot using machine learning and computer vision. *SN Computer Science*, 2021, vol. 2, no. 4, art. 251. DOI: 10.1007/s42979-021-00640-6.
19. Dli M.I., Lobaneva E.I., Puchkov A.Yu. Algorithms for the formation of images of the states of objects for their analysis by deep neural networks. *Journal of Applied Informatics*, 2019, vol. 2, pp. 43–55 (in Russ.).

Для цитирования

Пучков А.Ю., Лобанева Е.И., Василькова М.А. Программа формирования стека методов искусственного интеллекта при решении прикладных задач // Программные продукты и системы. 2021. Т. 34. № 3. С. 390–398. DOI: 10.15827/0236-235X.135.390-398.

For citation

Puchkov A.Yu., Lobaneva E.I., Vasilkova M.A. The program for forming a stack of artificial intelligence methods to solve applied problems. *Software & Systems*, 2021, vol. 34, no. 3, pp. 390–398 (in Russ.). DOI: 10.15827/0236-235X.135.390-398.