

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Ульяновский государственный технический университет»
Российская ассоциация искусственного интеллекта

**СЕМНАДЦАТАЯ
НАЦИОНАЛЬНАЯ КОНФЕРЕНЦИЯ
ПО ИСКУССТВЕННОМУ ИНТЕЛЛЕКТУ
С МЕЖДУНАРОДНЫМ УЧАСТИЕМ**

КИИ-2019

(21-25 октября 2019, г. Ульяновск, Россия)

Сборник научных трудов

Том 2

УЛЬЯНОВСК
УлГТУ
2019

УДК 004.8+004.89+004.82+004.032.26(045)

ББК 32.813(2А/Я)я43

С30

Редакционная коллегия: Н.Г. Ярушкина (ответственный за выпуск),
В.С. Мошкин

Организаторы конференции:

Российская ассоциация искусственного интеллекта

ФГБОУ ВО «Ульяновский государственный технический университет»

ФГБУ Федеральный исследовательский центр «Информатика и
управление» РАН

ФГБУН Институт проблем управления им. В.А. Трапезникова РАН

**Семнадцатая Национальная конференция по искусственному
интеллекту с международным участием. КИИ-2019** (21–25 октября
2019 г., г. Ульяновск, Россия). Сборник научных трудов. В 2 т. –
Ульяновск: УлГТУ, 2019. –

ISBN 978-5-9795-1938-8

Т.2. – 2019. – 227 с.

ISBN 978-5-9795-1940-1

Во втором томе трудов публикуются доклады участников конференции,
представленные на следующих секциях:

Секция 7. «Машинное обучение».

Секция 8. «Инженерия знаний и онтологии».

Секция 9. «Нейросетевые технологии».

Секция 10. «Интеллектуальный анализ данных».

Секция 11. «Моделирование рассуждений».

Конференция проводится при финансовой поддержке: Ульяновского
государственного технического университета, Российского фонда
фундаментальных исследований (проект № 19-47-731002), Фонда
развития информационных технологий Ульяновской области.

ISBN 978-5-9795-1938-8

ISBN 978-5-9795-1940-1

© Коллектив авторов, 2019

© Оформление, УлГТУ, 2019

ПРЕДИСЛОВИЕ

21-25 октября 2019 г. на базе Ульяновского государственного университета при поддержке Российской ассоциации искусственного интеллекта, ФИЦ «Информатика и управление» РАН и Института проблем управления им. В.А. Трапезникова РАН, состоится Семнадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2019.

Традиционно тематика конференции включает такие актуальные научные проблемы, как интеллектуальный анализ данных, анализ естественного языка, моделирование рассуждений, когнитивные исследования, мультиагентные системы, интеллектуальные роботы, машинное обучение, приложения методов искусственного интеллекта и другие.

В конференции предполагается участие более 150 исследователей в области искусственного интеллекта из Москвы, Санкт-Петербурга, Ульяновска, Калининграда, Твери, Смоленска, Нижнего Новгорода, Белгорода, Таганрога, Ростова-на-Дону, Самары, Саратова, Казани, Тольятти, Ярославля, Уфы, Екатеринбурга, Томска, Красноярска, Новосибирска, Иркутска, Хабаровска, Владивостока, а также зарубежных исследователей.

На конференцию было подано более 120 докладов. Согласно объявленным правилам, все доклады были поданы на русском языке. Кроме того, авторы, пожелавшие опубликоваться в сборнике серии «Communications in Computer and Information Science» издательства Springer, представили английские версии этих докладов. Все присланные работы прошли рецензирование членами Национального программного комитета, в результате которого было отобрано 75 докладов, принятых к представлению на конференции.

В настоящем сборнике публикуются:

- полные тексты принятых докладов, авторы которых не представили англоязычные версии;
- полные тексты принятых докладов, англоязычные версии которых по различным причинам не вошли в сборник издательства Springer;
- аннотации принятых докладов, англоязычные версии которых вошли в сборник издательства Springer.

Помимо пленарных и секционных докладов, в рамках конференции планируется проведение семинара совместно с представителями IT-сообщества Ульяновской области с целью выработки решений по кооперации и взаимному сотрудничеству в области применения современных фундаментальных технологий искусственного интеллекта в прикладных проектах IT-компаний.

Национальная конференция КИИ–2019 должна стать основой для плодотворной работы и поддержания эффективного профессионального сотрудничества исследователей в области искусственного интеллекта различных регионов России, а также зарубежных коллег.

Программный комитет КИИ-2019

Со-председатели КИИ 2019

Соколов И.А., академик РАН, Россия — ФИЦ ИУ РАН;

Васильев С.Н., академик РАН, Россия – ИПУ РАН;

Осипов Г.С., д.ф.-м.н., проф., Россия – ФИЦ ИУ РАН.

Председатель национального программного комитета

Кузнецов О.П., д.т.н., проф., ИПУ РАН, г. Москва.

Со-председатели Международного программного комитета

Vadim Stefanuk, Institute for Information Transmission Problems,
Russia;

Sergei Kuznetsov, National Research University Higher School of
Economics, Russia;

Aleksandr I. Panov, Russia — FRC «Computer Science and Control»
RAS.

Председатель организационного комитета КИИ-2019:

Ярушкина Н.Г., д.т.н., проф., УлГТУ, г. Ульяновск.

УДК 519.684.6

**ВЫЧИСЛЕНИЕ СХОДСТВ ОБУЧАЮЩИХ
ПРИМЕРОВ ДЛЯ ВКФ-МЕТОДА НА ВИДЕОКАРТАХ ¹**Д.В. Виноградов (*krrrguest@yandex.ru*)Федеральный исследовательский центр «Информатика и
управление» РАН, МоскваРоссийский Государственный Гуманитарный Университет,
Москва

В работе описывается реализация вероятностного алгоритма поиска сходств между обучающими примерами для вычислений на видеокартах общего назначения (GPGPU). Алгоритм был запрограммирован на языке OpenCL и исследовались его возможности с использованием видеокарты AMD Radeon VII под Kubuntu Linux 18.04 LTS.

Ключевые слова: сходство, вероятностный алгоритм, GPGPU, OpenCL

Введение

В работе [Vinogradov, 2018] был описан подход к машинному обучению, развивающему идеи ДСМ-метода автоматического порождения гипотез [Финн и др., 2009]. Ключевым нововведением являлись вероятностные алгоритмы вычисления некоторого подмножества сходств между обучающими примерами. Необходимость перейти на точку зрения статистического машинного обучения [Вапник и др., 1974] объясняется, с одной стороны, потенциально экспоненциальным числом различных сходств, что делает вычислительно сложным их исчерпывающее порождение, а с другой стороны — эффектом переобучения (появлением «фантомных» сходств), обнаруженным и исследованным автором [Виноградов, 2018a] ранее.

Вероятностные алгоритмы порождения сходств, исследованные ранее

¹ Работа выполнена при частичной поддержке гранта РФФИ № 18-29-03063мк.

и подробно описанные в [Vinogradov, 2018], основаны на различных вариантах цепей Маркова, осуществляющих спаренные блуждания по решётке всех сходств.

Хотя алгоритмы, основанные на спаривающих цепях Маркова, допускают хорошее распараллеливание (из-за независимости траекторий), возник вопрос о существовании вероятностного алгоритма, который может быть хорошо распараллеливаться для вычислений на графических картах общего назначения (GPGPU - General Purpose Graphical Processor Unit). В настоящее время подавляющее большинство многоядерных центральных процессоров (CPU) располагает 4-8 ядрами, тогда как графические карты фирм AMD и nVidia могут вести вычисления одновременно на 4096 Processing Elements (PE) — векторных вычислительных единицах. При этом процессорные единицы (объединённые в «волновой фронт») могут чередовать выполнение до 10 нитей (фронт), чтобы компенсировать задержки обращения к памяти.

1. Новый алгоритм поиска сходств

1.1. Мотивировка отказа от спаривающих цепей Маркова

Основной проблемой для вычислений на GPGPU является малый объем локальной памяти видеокарты, из-за чего необходимо подгружать данные из (глобальной) памяти компьютера. Случай алгоритмов, основанных на спаривающих цепях Маркова, является плохим в том смысле, что номер следующего обрабатываемого обучающего примера определяется датчиком случайных чисел и не может быть предсказан и загружен в локальную память заранее.

В статье [Виноградов, 2017] автор смог получить результат о достаточном числе порождаемых сходств в духе классической теоремы В.Н. Вапника и А.Я. Червоненкиса [Вапник и др., 1974]. Он гласит, что если для описания обучающих и тестовых (предъявленных для прогнозирования) примеров используется n бинарных атрибутов, то для того, чтобы с вероятностью, не меньшей $1 - \delta$, все ε -важные положительные тестовые объекты были предсказаны правильно, достаточно вероятностно породить $N \geq (2(n + 1) - 2 \lceil \log \delta \rceil) / \varepsilon$ сходств. Здесь важность тестового примера определяется как вероятность порождения хотя бы одного сходства, которое вкладывается в этот пример (и служит причиной положительного предсказания).

При анализе этого результата обращает на себя внимание независимость этой нижней границы от числа обучающих примеров

(объёма обучающей выборки). Это особенно интересно в связи с тем фактом, что на практике объём обучающей выборки может на несколько порядков превышать размерность признакового пространства (числа признаков, используемых для описания объектов).

Описываемый результат не зависит от вероятностного алгоритма для порождения сходств, что открывает возможность поручить порождение одного сходства каждой вычислительной единице GPGPU (а их количество может достигать 4096) или некоторой рабочей группе, если из-за слишком большой длины описание объектов не влезает в приватную память каждой рабочей единицы.

1.2. Краткое описание алгоритма

Обучающие примеры подаются на вход алгоритма в некотором фиксированном порядке. Тогда некоторый пакет этих примеров может быть загружен в локальную память GPGPU, и они могут обрабатываться рабочими группами (подмножествами вычислительных устройств внутри GPGPU) по порядку. Из главной теоремы статьи [Виноградов, 2017] можно оценить объём сходств, вычисляемых для достижения нужного уровня надёжности предсказания. При появлении нового обучающего примера рабочая группа, которая работает над одним из сходств, с некоторой вероятностью (зависящей только от самого примера) пересекает ранее вычисленное сходство с вновь поступившим примером.

Ключевой проблемой является установление факта, что вероятность порождения сходства описанным способом не зависит от порядка предъявления обучающих примеров. В работе [Виноградов, 2019] этот факт доказан. Доказательство опирается на то, что любая перестановка являет произведением транспозиций соседних элементов. Случай запрета контр-примеров, используя правило условной вероятности, сводится к случаю, когда контр-примеры не рассматриваются. Последний случай доказывается разбором различных ситуаций равенств и строгих включений в цепочках $h_3 \subseteq h_1 \subseteq h$ и $h_3 \subseteq h_2 \subseteq h$, где h — исходное сходство, o_1 и o_2 — обучающие примеры, $h_1 = h \cap o_1$, $h_2 = h \cap o_2$ и $h_3 = h \cap o_1 \cap o_2 = h_1 \cap o_2 = h_2 \cap o_1$.

2. Описание программы

Автор создал консольное приложение под Kubuntu 18.04 LTS, которое продемонстрировало высокое быстродействие предложенного им алгоритма. Выбор операционной системы определяется тем, что под Ubuntu 18.04 LTS имеется сконфигурированная сообществом

программистов платформа ROCm высокопроизводительных вычислений на GPGPU AMD (с дополнением к среде разработки Visual Studio Code). Впрочем, все компоненты ROCm имеют свободно распространяемые исходные тексты, что позволяет воссоздать её под любой Linux дистрибутив. Одной из компонент платформы ROCm является компилятор языка OpenCL C 2.0. Этот язык является открытым стандартом для создания программ для GPGPU, расширяя язык программирования C99.

Следует отметить, что современные GPGPU фирмы AMD (и nVidia) имеют внутренние регистры и ячейки памяти, допускающие хранение векторных типов (целочисленных и с плавающей запятой). Максимальная размерность поддерживаемого вектора достигает 16. Так как мы упаковываем наши битовые последовательности в `uint` (целое число без знака), то `uint16` может одновременно хранить (и, следовательно, PE обрабатывать за один такт) 512 бит. Этим достигается векторное распараллеливание нашего алгоритма (в дополнение к распараллеливанию по различным вычислительным элементам).

Если же длина описания обучающих примеров (и, как следствие, длина описания сходства) превосходит 512 бит, то можно использовать целую рабочую группу (содержащую 256 PE) для обработки одного сходства, так как локальная память у них общая. В этом случае обучающие примеры (и их сходства) могут иметь длину до 131 072 бит.

При превышении этого числа придётся немного изменить структуру хранимой информации (хранить дополнительно датчики случайных чисел). Но пока такая возможность не реализована.

Разработанная программа загружает массив обучающих примеров в локальную память, упаковывает их в `uint16` и, используя внутренний датчик случайных чисел, вычисляет сходства по основному алгоритму, аналогично упаковываются контр-примеры и проверяется «Запрет контр-примеров» и, наконец, упаковываются примеры, предъявленные для прогнозирования целевого свойства (съедобности гриба) и производится предсказание по аналогии. Для просмотра сходств они распаковываются из `uint16` в удобный вид.

3. Результаты экспериментов

Проводились эксперименты с видеокартой AMD Radeon VII. Эта видеокарта является аналогом профессиональной GPGPU AMD Instinct MI50, которая предназначена для высокопроизводительных вычислений на серверах. AMD Radeon VII имеет 3840 PE (60 CU x 4 vALU/CU x

16 PE/vALU). Она снабжена HBM2-видеопамятью объёма 16 Гигабайт и 4096 полосной шиной доступа к ней.

Эксперименты проводились на массиве Mushrooms из репозитория данных для тестирования алгоритмов машинного обучения Университета Калифорнии в Ирвайне - данных из определителя грибов Северной Америки [Lincoff, 1981], оцифрованных в файл `agaricus-lepota.data`

Этот массив содержит описания 8124 грибов, разделённых на две категории (съедобные и ядовитые). Мы случайным образом разделили их на обучающую и тестовую выборки. Обучающая выборка включает в себя 4032 объекта, из которых 2088 (+)-объектов (съедобные грибы). Тестовая выборка содержит 2120 (+)- и 1972 (-)-примеров (ядовитые грибы).

Каждый пример описывался 22 признаками, описывающие различные характеристики грибов (цвет, форма шляпки, места произрастания, частота встречаемости и т.п.). Эти признаки номинальные, принимающие одно из нескольких значений. ВКФ-система закодировала эти признаки битовыми строками длины 107 бит. При таком кодировании использовался алгоритм, предложенный и исследованный в работе [Виноградов, 2018b].

В ходе экспериментов вычислялось 61440 сходств 2088 обучающих примеров (съедобных грибов из обучающей части), некоторые из них впоследствии были отброшены с учётом «запрета контр-примеров», которыми являлись ядовитые грибы из обучающей выборки. Затем 4092 примера, составлявшие тестовую часть, доопределялись по аналогии с обучающими примерами с использованием уцелевших сходств.

Засекалось время порождения сходств на GPGPU. Оно оказалось не превосходящим 20 миллисекунд. Точность предсказания нового варианта машинного обучения достигла 100%. Впрочем, этот результат не является удивительным: в классическом (для CPU) подходе достаточно порождения от 80 до 150 сходств, прошедших через «запрет контр-примеров», чтобы обеспечить абсолютную точность предсказания.

Заключение

В работе описаны результаты исследования нового вероятностного алгоритма поиска сходств между обучающими примерами для вычислений на видеокартах общего назначения (GPGPU). Этот алгоритм был запрограммирован на языке OpenCL и запускался на видеокарте AMD Radeon VII под Kubuntu Linux 18.04 LTS. Результаты выглядят очень обнадеживающе.

Список литературы

[Вапник и др., 1974] Вапник В.Н., Червоненкис А.Я. *Теория распознавания образов (статистические проблемы обучения)*. - М.: Наука, Гл. ред. физ.-мат. Лит. - 1974. - 416 с.

[Виноградов, 2017] Виноградов Д.В. Надёжность предсказания по аналогии // *Научно-техническая информация (Серия 2)*, 2017, № 7, С. 11-15; Vinogradov D.V. The reliability of analogy-based prediction // *Automatic Documentation and Mathematical Linguistics*. – 2017. – Vol. 51, № 4. – P. 191-195.

[Виноградов, 2018a] Виноградов Д.В. Скорость сходимости к пределу вероятности порождения случайного сходства при наличии контр-примеров // *Научно-техническая информация (Серия 2)*, 2018, № 2, С. 21-24; Vinogradov D.V. The Rate of Convergence to the Limit of the Probability of Encountering an Accidental Similarity in the Presence of Counter Examples // *Automatic Documentation and Mathematical Linguistics*. – 2018. – Vol. 52, № 1. – P. 35-37.

[Виноградов, 2018b] Виноградов Д.В. О представлении объектов битовыми строками для ВКФ-метода // *Научно-техническая информация (Серия 2)*, 2018, № 5, С. 1-4; Vinogradov D.V. On Object Representation by Bit Strings for the VKF-Method // *Automatic Documentation and Mathematical Linguistics*. – 2018. – Vol. 52, № 3. – P. 113-116.

[Виноградов, 2019] Виноградов Д.В. Ещё один вероятностный алгоритм для вычисления сходств // *Научно-техническая информация (Серия 2)*, 2019 (в печати).

[Финн и др., 2009] ДСМ-метод автоматического порождения гипотез: Логические и эпистемологические основания. (ред.: Финн В.К., Аншаков О.М.) - М.: Эдиториал УРСС - 2009. - 432 с.

[Lincoff, 1981] Lincoff, G.H. *The Audubon Society Field Guide to North American Mushrooms*. - NY: Knopf, 1981. - 926 pp.

[Vinogradov, 2018] Vinogradov D.V. Machine Learning Based on Similarity Operation // Proc. 16th Russian Conference on AI (RCAI 2018). Communications in Computer and Information Science. – 2018. – Vol. 934. – P. 46-59

ПЕРЕОБУЧЕНИЕ В ДСМ-СИСТЕМАХ

Д.А. Добрынин (minirobot@yandex.ru)

РГГУ, Москва

ФИЦ ИУ РАН, Москва

В работе описываются эффекты переобучения, возникающие при использовании ДСМ-систем, и методы борьбы с ними. В качестве примеров приводятся данные по ДСМ-системам прогнозирования биологических свойств химических соединений и прогнозированию медицинских данных.

Ключевые слова: интеллектуальные системы, ДСМ-метод, переобучение

Введение

В настоящее время интеллектуальный анализ данных нашел широкое применение во многих практически значимых областях, таких как фармакология, медицина, социология и многих других. Интеллектуальные системы, построенные с использованием ДСМ-метода [Финн 2009], успешно используются для предсказания биологической активности химических соединений, медицинской диагностики, социологических исследований и др.

При проведении компьютерных экспериментов с помощью ДСМ-систем существуют две стадии обработки данных: обучение на основе примеров из базы фактов и предсказание ранее неизвестных примеров. На ранних этапах использования ДСМ-систем считалось, что для более правильного и полного доопределения тестовых примеров необходимо иметь как можно большую базу фактов и порождать все возможные сходства этих примеров. Далее методы фальсификации гипотез, используемые ДСМ-методом, позволят отобрать из порожденного множества гипотез «хорошие» и отбросить «плохие». Считалось, что тем самым можно получить качественную базу гипотез, которые и обеспечат наилучшее предсказание тестовых примеров.

При практическом применении ДСМ-метода наблюдаются два интересных феномена. При увеличении объема базы фактов, т.е. количества примеров, наблюдается эффект комбинаторного взрыва, присущий всем логическим методам анализа данных. Этот эффект заключается в том, что при линейном росте количества примеров количество получаемых гипотез растет экспоненциально. Такой рост

объясняется тем, что гипотезы являются общими частями всех примеров и основная операция получения гипотез – пересечение выполняется по принципу «каждый с каждым», что имеет экспоненциальный характер роста. Такой взрывной рост количества гипотез ограничивает объемы данных, используемые для обучения, поскольку для получения гипотез требуется все больше и больше компьютерных ресурсов (памяти) и затрачивается все больше времени.

Другим эффектом, проявляющимся не так явно, как комбинаторный взрыв, является эффект переобучения. Этот эффект малоизучен для ДСМ-систем, поскольку количество данных, представляемых экспертами для компьютерных экспериментов, всегда ограничено.

1. Переобучение в задачах искусственного интеллекта

Переобучение (англ. overfitting) в машинном обучении и статистике — явление, когда построенная модель хорошо объясняет примеры из обучающей выборки, но относительно плохо работает на примерах, не участвовавших в обучении (на примерах из тестовой выборки).

Это связано с тем, что при построении модели («в процессе обучения») в обучающей выборке обнаруживаются некоторые случайные закономерности, которые отсутствуют в примерах тестовой выборки.

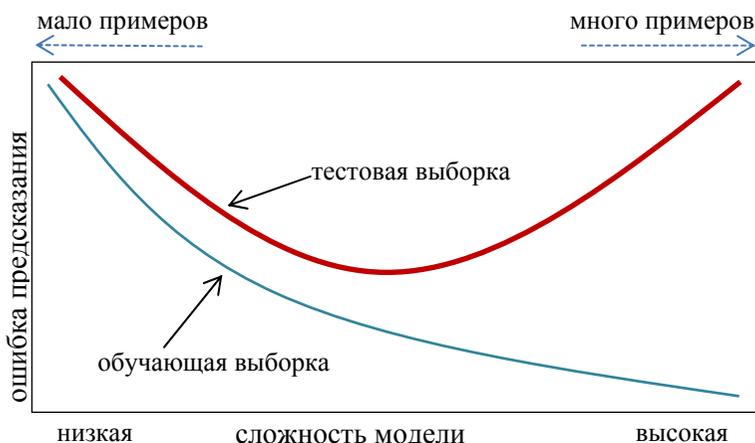


Рис. 1 Зависимость ошибок предсказания от сложности модели

На рис. 1 представлена типичная зависимость ошибки предсказания от сложности построенной модели. Для ДСМ-систем сложность модели

пропорциональна количеству полученных гипотез, т.е. экспоненциально зависит от количества обучающих примеров.

При низкой сложности модели (малом количестве обучающих примеров) наблюдается устойчивое уменьшение ошибки предсказания как на обучающей выборке, так и на тестовых примерах. Эту область можно назвать областью недообучения, когда полезной информации, извлеченной из обучающей выборки недостаточно для качественного предсказания.

При дальнейшем росте сложности модели ошибки предсказания на обучающей выборке продолжают уменьшаться, а ошибки на тестовой выборке входят в область локального минимума и перестают уменьшаться. Эта область считается наиболее оптимальной для прогноза, поскольку в ней ошибки предсказания на тестовой выборке минимальны. Эту область можно назвать областью оптимального обучения.

При дальнейшем росте сложности модели (большом количестве примеров) ошибки предсказания на обучающей выборке продолжают уменьшаться. Однако ошибки для тестовой выборки начинают расти. Эта область называют переобучением. Для этой области эффективность модели на новых данных будет ниже, чем на данных, использовавшихся для обучения. При переобучении в ДСМ-системах в результате пересечения обучающих примеров появляется большое количество «шумовых» гипотез, которые уменьшают точность предсказания тестовых примеров.

Способы борьбы с переобучением в задачах искусственного интеллекта зависят от метода моделирования и способа построения модели. Например, если строится дерево принятия решений, то можно обрезать некоторые его ветки в процессе построения. Для обучения нейронных сетей используют некоторое закругление входных данных [Bramer 2007].

2. Методы борьбы с переобучением в ДСМ-системах

Для борьбы с переобучением в ДСМ-системах использовались различные способы, общей идеей которых было сокращение гипотез.

Одним из самых эффективных способов уменьшения «плохих» гипотез является применение ДСМ-метода с запретом на контрпримеры [Финн 2009] и аналогов этого метода. Идея метода заключается в том, что полученная пересечением обучающих примеров гипотеза проверяется на вхождение в противоположные по знаку родители (обучающие примеры). Так полученные после пересечения положительные гипотезы проверяются на вхождение в отрицательные примеры. Если такая

гипотеза является частью отрицательного примера, то она отбрасывается. Аналогично полученные отрицательные гипотезы проверяются на вхождение в положительные примеры.

Использование ДСМ-метода с запретом на контрпримеры позволяет не просто уменьшить общее количество гипотез, но и отбросить большую часть противоречивых гипотез. Например, при прогнозировании мутагенной активности нитрозамещенных бензолов [Харчевникова 2015] всего исследовалось 254 соединения. Для 176 положительных примеров и 78 отрицательных ДСМ-метод без запрета на контрпримеры дает 247 положительных и 58 отрицательных гипотез. ДСМ-метод с запретом на контрпримеры дает 130 положительных и 21 отрицательных примеров. При этом результаты теста доопределения по одному для метода с запретом на контрпримеры дает большую точность, чем для метода без запрета на контрпримеры.

При прогнозе биологической активности химических соединений с помощью ДСМ-систем метод с запретом на контрпримеры является основным, поскольку при несколько меньшей полноте прогноза дает существенно большую точность.

Для медицинских приложений характерно большое количество признаков, входящее в один исходный пример. Например, при анализе заболевания панкреатитом [Винокурова 2015] для описания одного примера использовался 41 признак, каждый из которых в свою очередь включал в себя до пятнадцати полей. Эффективным методом уменьшения количества гипотез для приложений с таким типом данных являлась фильтрация гипотез по количеству родителей. В табл. 1 представлена зависимость количества положительных и отрицательных гипотез для эксперимента с прогнозированием панкреатита. Всего в эксперименте было 12 положительных и 10 отрицательных примеров (родителей). Гипотезы, которые содержали в своей структуре количество родителей, менее заданного числа отбрасывались.

Таблица 1

Порог	Кол-во (+) гип.	Кол-во (-) гип.
2	103	31
3	60	8
4	20	3
5	4	1
6	2	0
7	1	0
8	0	0

Анализ зависимости количества гипотез от порогового количества родителей дает резкий спад – примерно в 10 раз при увеличении порога в два раза. При некотором значении порога, в данном эксперименте 8 для положительных примеров и 6 для отрицательных, наблюдается исчезновение гипотез. Оптимальными значениями порога для данного эксперимента были значения 6 для положительных и 4 для отрицательных примеров. При этом было получено две положительные и три отрицательных гипотезы, которые обеспечили наилучшее предсказание в данном эксперименте.

С помощью регулирования порога количества родителей удается эффективно снижать общее количество гипотез. Недостатком данного способа является тот факт, что порог для положительных и отрицательных примеров приходится искать отдельно для каждого эксперимента, что является довольно трудоемким процессом.

Другой способ борьбы с переобучением для ДСМ-систем - это ограничение количества исходных примеров. Этот способ подходит для тех случаев, когда количество примеров велико, а количество признаков для одного примера сравнительно невелико. Такие типы данных используют ДСМ-системы для прогнозирования биологических свойств химических соединений. Например, при прогнозировании мутагенной активности нитрозамещенных бензолов [Харчевникова 2015] всего исследовалось 254 соединения, из них было сформировано 176 положительных и 78 отрицательных примеров. В табл. 2 представлена зависимость количества положительных гипотез от количества родителей.

Анализ зависимости количества гипотез от количества родителей показывает примерно линейную зависимость. Так увеличение количества родителей в два раза дает примерно двукратное увеличение количества гипотез. Такая зависимость характерна для тех областей, где в описании исходных примеров наблюдается сравнительно небольшое количество признаков. Для вышеупомянутого эксперимента каждый пример описывался множеством от двух до двадцати признаков, что существенно меньше, чем для медицинских приложений.

Таблица 2

Кол-во родителей	Кол-во гипотез	Кол-во родителей	Кол-во гипотез	Кол-во родителей	Кол-во гипотез
10	4	70	76	130	168
20	14	80	88	140	191
30	20	90	103	150	209
40	31	100	120	160	222
50	47	110	136	170	239
60	63	120	151	176	247

Для медицинских приложений, где количество признаков для одного примера велико наблюдается резкий рост количества получаемых гипотез от количества примеров (эффект комбинаторного взрыва). Эффективным способом борьбы с переобучением в ДСМ-системах является устранение незначимых признаков из описания примеров. При первичном подборе признаков для медицинских приложений в исходный набор вносятся все возможные признаки, описывающие состояние больного. Фактически все содержимое карты больного переносится в электронную форму описания пациента. Следует ожидать, что большая часть такого рода информации не будет нести полезный смысл для целевого прогноза. Трудность заключается в том, что заранее не известно, какие признаки окажутся информативными, а какие – нет.

Эффективным способом исключения неинформативных признаков являлось применение следующей методики [Панкратова 2014]. При проведении экспериментов по медицинской диагностике все признаки делились три группы:

- 1) группа необходимых признаков – признаки, без наличия которых гипотеза не имеет смысла;
- 2) группа существенных признаков – признаки, без наличия хотя бы одного из которых, гипотеза не имеет смысла;
- 3) группа сопутствующих признаков – признаки не входящие в группы 1) и 2), например, пол, возраст.

Формулировалось дополнительное требование к гипотезам:

«В потенциальную гипотезу должны входить все признаки группы (1) и хотя бы один из признаков группы (2)».

Для отсеивания признаков было введено понятие двух фильтров – конъюнктивного и дизъюнктивного. Конъюнктивный фильтр требует вхождения в потенциальную гипотезу всех перечисленных в нем признаков. Дизъюнктивный фильтр требует включения в потенциальную гипотезу хотя бы одного из перечисленных в дизъюнктивном фильтре признаков.

Подбор признаков, которые должны были входить в тот или иной фильтр осуществлялся вручную по результатам скользящего контроля [Винокурова 2015]. Использование фильтров позволило не просто сократить общее количество гипотез, но и выделить среди них информативные. Было сокращено 11 неинформативных признаков из 60 признаков, участвующих в описании.

В настоящее время ведутся исследования по автоматизированному подбору информативных признаков, но успехи в этом направлении пока скромные.

Одним из методов борьбы с переобучением в ДСМ-системах является получение только части от всего множества гипотез. Если этой части будет достаточно для успешного предсказания тестовых примеров, то нет необходимости вычислять все гипотезы, достаточно найти случайное подмножество сходств.

В работе [Виноградов 2017] описывает метод получения случайного набора гипотез с помощью спаривающихся цепей Маркова (ВКФ-метод). Спаривающая цепь Маркова используется для порождения ВКФ гипотез из обучающих примеров, при этом независимые траектории этой цепи порождают независимые элементы решетки кандидатов в гипотезы.

Как показали эксперименты на реальных данных, отказ от абдуктивного уточнения за счет увеличения объема порождаемых ВКФ-гипотез на этапе индукции (получения гипотез) приводит к уменьшению общего количества ВКФ-гипотез на несколько порядков (и соответствующему сокращению времени работы) без снижения качества предсказания тестовых примеров. Применять ВКФ-метод получения гипотез можно на данных с большим количеством обучающих примеров, где порождается избыточное количество гипотез.

Заключение

В настоящее время в современных ДСМ-системах эффект переобучения возникает из-за экспоненциального роста количества гипотез при увеличении количества обучающих примеров. Борьба с этим явлением требует учета свойств используемой для прогноза информации о самих примерах и определяющих их признаках.

Для ДСМ-систем медицинской диагностики характерно сравнительно небольшое количество исходных примеров (каждый пример представляет собой описание больного) и большое количество признаков, которое может доходить до нескольких сотен. Рост количества получаемых гипотез при возрастании количества примеров является близким к экспоненциальному. Эффективными способами борьбы с переобучением в данном случае являются применение ограничения на количество родителей и устранение незначимых признаков с помощью фильтров. Ограничивать количество примеров в данном случае не удается, поскольку самих обучающих примеров довольно мало.

Для ДСМ-систем предсказания биологических свойств химических соединений характерно большое количество исходных примеров и сравнительно небольшое количество признаков для описания одного примера. Рост количества гипотез при возрастании количества примеров

является близким к линейному. Наиболее эффективным способом уменьшения количества неинформативных гипотез является использование ДСМ-метода с запретом на контрпримеры. Применение ограничения на количество родителей в данных системах является малоэффективным, поскольку максимальное количество родителей у одной гипотезы не превышает 8-10 при общем количестве примеров в массиве порядка 300.

Эффективным способом борьбы с переобучением в данном случае является варьирование количества обучающих примеров. Поскольку признаками для систем данного вида являются дескрипторы языков представления структуры химических соединений, то исключение неинформативных признаков в данном случае является нетривиальной задачей. Практика показывает, что изменение способа кодирования химических соединений сильно влияет на количество гипотез, их информативность и на результаты прогноза.

Автоматизированный подбор информативных признаков является важным направлением исследования, который позволит улучшить прогнозирование для медицинских приложений ДСМ-систем, а также оценить эффективность представления данных для систем предсказания биологической активности химических соединений.

Список литературы

[Виноградов 2017] Виноградов Д.В. Эффективность ленивых вычислений для поиска сходств в ВКФ-системе // Научная и техническая информация, Сер. 2. – 2017. – № 4. – С. 19–23

[Винокурова 2015] Винокурова Л.В., Агафонов М.А., Варванина Г.Г., Финн В.К., Панкратова Е.С., Добрынин Д.А. Применение интеллектуальной системы типа ДСМ для анализа клинических данных у больных хроническим панкреатитом//Экспериментальная и клиническая гастроэнтерология. 2015. № 5 (117). С. 81-а.

[Панкратова 2014] Панкратова Е.С., Добрынин Д.А. Об одном способе выявления неинформативных признаков в интеллектуальных ДСМ-системах по медицине//НТИ Серия 2: Информационные процессы и системы. 2014. № 3. С. 10-13.

[Финн 2009] Автоматическое порождение гипотез в интеллектуальных системах. Сост. Е.С. панкратова, В.К. Финн, под общ. Редакцией В.К. Финна – М.: Книжный дом «Либроком», 2009.

[Харчевникова 2015] Харчевникова Н.В., Блинова В.Г., Добрынин Д.А., Журков В.С. Использование интеллектуальной ДСМ-системы для анализа связи структуры нитрозамещенных бензолов с их мутагенной активностью в тесте Эймса//НТИ Серия 2: Информационные процессы и системы. 2015. № 3. С. 6-11.

[Bramer 2007] Max Bramer Principles of Data mining/Springer, 2007.

УДК 681.518.5

ОБНОВЛЕНИЕ АГРЕГИРОВАННЫХ КЛАССИФИКАТОРОВ ПРИ ПОСТУПЛЕНИИ НОВОЙ ИНФОРМАЦИИ О ПОКАЗАТЕЛЯХ ФУНКЦИОНИРОВАНИЕ ОБЪЕКТА

Д.А. Жуков (*zh.dimka17@mail.ru*)

Ульяновский филиал ПАО «Туполев», Ульяновск

В.Н. Клячкин (*v_kl@mail.ru*)

В.Р. Крашенинников (*kvrulstu@mail.ru*)

Ю.Е. Кувайскова (*u.kuvaiskova@mail.ru*)

Ульяновский государственный технический университет,
Ульяновск

В работе рассмотрена методика построения агрегированного классификатора для диагностики функционирования технического объекта. По истечении некоторого времени по мере поступления новых показателей работы модель классификатора устаревает, и ее необходимо обновить. Рассмотрено псевдоградиентное обновление параметров базовых моделей, входящих в состав агрегата. Однако в дальнейшем необходимо обновление и структуры модели.

Ключевые слова: агрегированные классификаторы, псевдоградиентное обновление параметров модели, обновление структуры модели

Введение

Проводится мониторинг показателей функционирования технического объекта в процессе эксплуатации с целью диагностики и прогнозирования его состояния [Биргер, 1978]. Задача может быть сведена к бинарной классификации – разделению состояний объекта на работоспособные или неработоспособные и решена методами машинного обучения. Как правило, используются нейронные сети, бэггинг деревьев решений, градиентный бустинг и другие методы [Witten et al., 2005].

Для увеличения точности прогнозирования предложен агрегированный подход – использование комбинации нескольких базовых методов классификации.

Например, работа системы водоочистки характеризуется восьмью показателями качества очистки воды: содержанием хлоридов, алюминия, окисляемостью, щелочностью и т.п. Если хотя бы один из показателей

выходит за допустимые пределы, система считается неисправной и требует настройки.

При вибромониторинге гидроагрегата контролировались десять показателей: вибрации нижнего генераторного подшипника верхнего бьефа и на правом берегу, соответствующие вибрации верхнего генераторного подшипника, бой вала гидротурбины и вала гидрогенератора, вибрации крышки гидротурбины. Выход вибраций за допустимые границы свидетельствует о нарушении работоспособности гидроагрегата.

По мере поступления новой информации о показателях функционирования объекта математическая модель классификации может устаревать: изменяются как структура модели (состав агрегированного классификатора), так и параметры составляющих базовых классификаторов.

Цель исследования – разработка алгоритма обновления агрегированных классификаторов для прогнозирования и диагностики состояния технического объекта по мере поступления новых данных о показателях его функционирования.

1. Базовые методы машинного обучения

Исходными данными для прогнозирования и диагностики состояния объекта являются матрица показателей функционирования системы X , элементы которой x_{ij} – результат i -го наблюдения по j -му показателю; $i = 1, \dots, l$, $j = 1, \dots, p$ (l – количество строк, или число наблюдений, p – количество столбцов, или число показателей функционирования), и вектор-столбец Y , состоящий из ответов 0 или 1, для опытов, где объект оказался неработоспособным и работоспособным соответственно. Совокупность (x_i, y_i) , где значение y_i вектора Y соответствует строке x_i матрицы X , представляет прецедент.

Часто вместо нуля и единицы используют множество $Y = \{-1; +1\}$. В этом случае модель алгоритма классификации имеет вид [Witten et al., 2005]:

$$a(x, w) = \text{sign} \sum_{j=0}^p w_j x_j \quad (x_0 = 1). \quad (1.1)$$

Расчеты проводились с применением библиотеки инструментов Statistics and Machine Learning Toolbox в пакете Matlab. Использовались 11 методов: логистическая регрессия, дискриминантный анализ, наивный байесовский классификатор, нейронные сети, метод опорных векторов, композиционные методы (бэггинг деревьев решений и

пять методов бустинга: градиентный бустинг, AdaBoost, LogitBoost, GentleBoost и RUSBoost [Neykov et al., 2016]).

Для оценки качества классификации применялась кросс-валидация, при этом критерием качества служила F -мера – гармоническое среднее точности и полноты: качество классификации тем выше, чем ближе значение F к единице [Davis et al., 2006].

Варьировался объем контрольной выборки (количество блоков кросс-валидации) и количество показателей функционирования: с использованием различных критериев отбирались значимые показатели.

При исследовании исправности системы водоочистки по восьми показателям качества воды (по результатам мониторинга 348 наблюдений, из них в 47 случаях имели место нарушения работы системы) лучший результат показал бэггинг деревьев решений, при котором F -мера оказалась 0,8578. Объем контрольной выборки составил 10 % (10 блоков при кросс-валидации), значимы шесть показателей из восьми (значимость оценивалась по критерию Стьюдента для линейной регрессионной модели).

В аналогичном исследовании вибростойкости гидроагрегата по 10 показателям вибродатчиков, установленных на различных участках гидроагрегата (1500 наблюдений, нарушения работы в 312 прецедентах), лучшим из базовых методов оказался градиентный бустинг с значением F -меры, равным 0,8923. При этом объем контрольной выборки составил 20%, значимыми оказались девять показателей вибродатчиков (показания одного из датчиков оказались сильно коррелированы с другим).

2. Применение агрегированных кластеров

Для повышения качества бинарной классификации эффективен агрегированный подход. Это было показано на примере кредитного скоринга и позднее применено к решению задач технической диагностики [Klyachkin et al., 2017].

В агрегированном подходе совместно используются различные базовые классификаторы, построенные на обучающей выборке. Для достижения наилучшего результата можно использовать разные методы объединения результатов: по среднему значению, по медиане, по голосованию.

Например, при агрегировании по среднему значению вероятность $\hat{P}_{AKcp}(X_r)$ того, что r -й объект работоспособен, определяется по формуле.

$$\hat{P}_{AKcp}(X_r) = \frac{\sum_{K=1}^H \hat{P}_K(X_r)}{H}, \quad (2.1)$$

где $\hat{P}_K(X_r)$ – вероятность того, что r -й объект работоспособен, найденная с помощью K -го базового метода, $K = 1, \dots, H$.

Возможен как полный перебор базовых методов (в общем случае число наборов равно $2^H - 1$, при $H = 11$ количество возможных ситуаций для каждого метода агрегирования составит 2047), так и частичный. При этом следует учесть два обстоятельства. Во-первых, на примерах диагностики реальных технических объектов показано, что наилучший из базовых классификаторов (обеспечивший максимум F -меры) всегда входит в состав агрегата. Во-вторых, увеличение числа компонентов агрегата больше двух не приводит к значимому увеличению F -меры. С учетом изложенного при $H = 11$ количество переборов сокращается до 10: агрегат состоит из двух компонент, один из которых – базовый классификатор, обеспечивающий максимум F -меры, в качестве второго перебираются оставшиеся 10 классификаторов.

Эти два факта требуют проверки для каждого конкретного объекта. Поскольку контрольная выборка формируется случайным образом, результаты классификации при повторении испытаний каждый раз несколько отличаются друг от друга. В частности, получаются и разные агрегаты, максимизирующие F -меру.

В таблице 1 частично приведены результаты испытаний системы водоочистки: отобраны десять первых испытаний, в которых агрегат включает два базовых компонента (правая часть таблицы) и такое же число испытаний с большим количеством компонент (левая часть таблицы: от трех до пяти компонент).

Анализируя полученные данные, видим, что во все агрегаты вошел, как компонент, бэггинг деревьев решений (этот классификатор оказался лучшим из базовых методов для рассматриваемого объекта).

Значения F -меры как по первому, так и по второму столбцу не противоречат предположению о нормальности распределения этой характеристики. Это предположение проверялось по критерию Шапиро-Уилка (для малых выборок).

Проверим гипотезу о равенстве значений F -меры в левой и правой частях таблицы 1. Вначале проверяется гипотеза о равенстве дисперсий по критерию Фишера (таблица 2). Видим, что эта гипотеза отвергается: выборочное значение критерия 3,34 больше критического 3,18.

Поэтому проверяем гипотезу о равенстве средних значений F -меры,

используя двухвыборочный тест Стьюдента с *различными* дисперсиями (таблица 3).

Таблица 1. Структуры классификаторов и F -мера

Структура классификатора	F -мера (F_1)	Структура классификатора	F -мера (F_2)
AM-C: GrB+ БДР+ LB	0,9077	AM-C: БДР+ GB	0,9061
AM-C: HC+ GrB+ БДР+ LB+ GB	0,9028	AM-C: БДР+ LB	0,9016
AM-C: ЛР+ БДР+ АВ	0,9067	AM-C: БДР+ АВ	0,9040
AM-C: МОВ+ ЛР+ БДР+ LB+ GB	0,9015	AM-M: БДР+ GB	0,9061
AM-M: GrB+ БДР+ LB	0,9049	AM-M: БДР+ АВ	0,9056
AM-M: ЛР+ БДР+ GB+ RB	0,8960	AM-M: БДР+ LB	0,9075
AM-M: ЛР+ БДР+ LB+ RB	0,9037	AM-G: БДР+ GB	0,9061
AM-G: GrB+ БДР+ LB	0,9044	AM-G: БДР+ LB	0,9004
AM-G: ЛР+ БДР+ АВ	0,9070	AM-G: БДР+ АВ	0,9040
AM-G: МОВ+ GrB+ БДР	0,8962	AM-G: БДР+ LB	0,9064

Обозначения: AM-C, AM-M, AM-G – агрегирование по среднему, по медиане и голосованию соответственно, GrB – градиентный бустинг, БДР – бэггинг деревьев решений, АВ – AdaBoost, МОВ – метод опорных векторов, LB – LogitBoost, ДА – дискриминантный анализ, RB – RUSBoost, GB – GentleBoost, ЛР – логистическая регрессия, HC – нейронная сеть.

Таблица 2. Двухвыборочный тест Фишера для дисперсии

	F_1	F_2
Среднее	0,9031	0,90478
Дисперсия	2E-05	5,15956E-06
Наблюдения	10	10
Число степеней свободы	9	9
Выборочное значение статистики Фишера	3,3442	
Квантиль распределения Фишера (критическое значение)	3,1789	

Таблица 3. Двухвыборочный t -тест с различными дисперсиями

	F_1	F_2
Среднее	0,90309	0,90478
Дисперсия	1,73E-05	5,15956E-06
Наблюдения	10	10
Гипотетическая разность средних	0	
Число степеней свободы	14	
Выборочное значение t -статистики	-1,12883	
t критическое одностороннее	1,76131	
t критическое двухстороннее	2,144787	

Выборочное значение статистики Стьюдента – 1,13 по модулю меньше критического значения (как для одностороннего, так и двустороннего критериев), что подтверждает выдвинутую гипотезу:

значение F -меры при двух компонентах агрегированного классификатора отличается от агрегата с большим числом компонент незначимо.

Заметим, что использование агрегированного классификатора для системы водоочистки в наилучшем варианте (агрегирование по среднему значению, сочетание градиентного бустинга, бэггинга деревьев решений и метода LogitBoost) привело к повышению F -меры с значения 0,8578 до 0,9077 (на 5,8 %).

Подобный результат получен при построении агрегированных классификаторов для оценки вибростойкости гидроагрегата: значение F -меры повысилось на 3,1 %: с 0,8923 до 0,9204 (агрегирование по медиане, сочетание градиентного бустинга и логистической регрессии).

На двух рассмотренных примерах реальных технических объектов была проведена проверка гипотезы о том, что агрегирование значимо увеличивает F -меру. Эта гипотеза не противоречит результатам опытов.

3. Обновление параметров базовых классификаторов

По истечении некоторого времени поступает новая информация о показателях функционирования объекта. Требуется создать способ обновления классификатора для адаптации к изменяющимся условиям, о

которых можно судить по вновь поступающим данным.

Адаптация полученных агрегированных классификаторов осуществляется путем корректировки параметров моделей, входящих в состав агрегированных классификаторов с использованием рекуррентной псевдоградиентной процедуры [Поляк и др., 1973], [Krashennnikov et al., 2017]. Данная процедура позволяет находить оптимальные оценки параметров модели с достаточно быстрой сходимостью при минимальных вычислительных затратах по сравнению с другими методами.

Оптимизация модели состоит в нахождении вектора её параметров, минимизирующих дисперсию ошибок или, что то же самое, сумму квадратов

$$\Omega(\bar{q}) = \sum_{r=1}^l (P(Y_r) - \hat{P}(X_r, \bar{q}))^2, \quad (3.1)$$

где $P(Y_r)$ - «истинная» вероятность принадлежности r -го объекта классу, $\hat{P}(X_r, \bar{q})$ - оценка вероятности принадлежности r -го объекта классу, зависящая от показателей \bar{X} объекта и вектора параметров модели \bar{q} .

Суть псевдоградиентной процедуры заключается в постоянной корректировке вектора параметров модели при поступлении информации о каждом новом объекте на основе следующей формулы.

$$\bar{q}_{r+1} = \bar{q}_r - \nu_r \nabla J(\bar{q}_r), \quad (3.2)$$

где \bar{q}_{r+1} - следующее за \bar{q}_r приближение оптимального вектора \bar{q} , ν_r - коэффициенты, влияющие на величину шага,

$$\nabla J(\bar{q}_r) = \nabla (P(Y_r) - \hat{P}(X_r, \bar{q}_r))^2$$

- градиент отдельного слагаемого из формулы (3.1), который является псевдоградиентом функции $\Omega(\bar{q})$.

Корректировка параметров модели достигается путем добавления к старому вектору параметров \bar{q}_r поправки, получаемой в результате умножения числа ν_r на псевдоградиент $\nabla J(\bar{q}_r)$. При этом важно, что эта процедура способна отслеживать меняющуюся ситуацию, так как текущий вектор \bar{q}_r постоянно на неё реагирует в соответствии с новыми показателями.

Рассмотрим применение псевдоградиентной процедуры обновления коэффициентов модели на примере одного из базовых классификаторов – модели логистической регрессии. Тогда $\hat{P}(X_r, \bar{q}_r)$ находится по формуле:

$$\hat{P}(X_r, \bar{q}_r) = f(z(\bar{q}_r)), \quad z(\bar{q}_r) = q_{r0} + q_{r1}x_{r1} + \dots + q_{rd}x_{rd}, \quad (3.3)$$

где $f(z(\bar{q}_r))$ – логистическая функция:

$$f(z(\bar{q}_r)) = \frac{1}{1 + e^{-z(\bar{q}_r)}}. \quad (3.4)$$

По определению

$$\nabla J(\bar{q}_r) = \frac{d}{d\bar{q}_r} (P(Y_r) - \hat{P}(X_r, \bar{q}_r))^2$$

Поэтому преобразуем (3.2):

$$\begin{aligned} \bar{q}_{r+1} &= \bar{q}_r - \nu_r \nabla J(\bar{q}_r) = \bar{q}_r - \nu_r \frac{d}{d\bar{q}_r} (P(Y_r) - \hat{P}(X_r, \bar{q}_r))^2 = \\ &= \bar{q}_r + 2\nu_r (P(Y_r) - \hat{P}(X_r, \bar{q}_r)) \frac{d\hat{P}(X_r, \bar{q}_r)}{d\bar{q}_r} = \bar{q}_r + \eta_r (P(Y_r) - \hat{P}(X_r, \bar{q}_r)) \frac{d\hat{P}(X_r, \bar{q}_r)}{d\bar{q}_r} \end{aligned}$$

где $\eta_r = 2\nu_r$ – новый параметр шага процедуры. Тогда корректировка коэффициентов логистической регрессии выполняется по формуле:

$$\bar{q}_{r+1} = \bar{q}_r + \eta_r (P(Y_r) - \hat{P}(X_r, \bar{q}_r)) \frac{d\hat{P}(X_r, \bar{q}_r)}{d\bar{q}_r}. \quad (3.5)$$

Предлагается следующий способ обновления параметров моделей прогнозирования на примере логистической регрессии:

1. За начальный вектор параметров логистической регрессии принимается оптимальный вектор этой модели, построенный на «старых» данных.

2. Для классификации каждого вновь поступающего набора данных применяется модель логистической регрессии с последним вектором её коэффициентов, имеющимся на момент поступления новой информации.

3. По мере поступления значений $P(Y_r)$ выполняется процедура корректировки параметров модели на основе псевдоградиента (3.5).

4. Обновление структуры агрегированных классификаторов

По истечении некоторого времени структура модели классификации может «устареть», и применение псевдоградиентной процедуры (которая оптимизирует только параметры модели) может оказаться недостаточным. В этом случае предлагается произвести обновление структуры модели.

Можно предложить различные подходы к обновлению структуры. Первый вариант таков. На основании опыта и с учётом рекомендаций специалистов производится регулярное обновление структуры модели через определённый период (например, один раз в месяц, в квартал и т.п.). Тогда проводится структурно-параметрическая идентификация модели по всем данным, полученным за этот период; а если этих данных

недостаточно, то используются и данные за предшествующий период.

Другой подход состоит в том, что при построении исходной структуры выбираются две или более конкурирующих моделей: не только та, для которой F -мера оказалась максимальной, но и предыдущие по качеству модели (они будут запасными). Вначале применяется лучшая модель, а качество запасных оценивается виртуально. С течением времени более хорошие результаты может показывать одна из запасных моделей, тогда она становится основной, то есть применяющейся на практике, а применяемая до этого модель переводится в запасные.

Заключение

В работе предложена методика построения агрегированного классификатора для диагностики функционирования технического объекта. По мере появления новой информации о показателях работы объекта возникает необходимость обновления модели классификации. Это можно сделать с помощью псевдоградиентной процедуры для обновления параметров базовых моделей, а при необходимости и обновления структуры агрегированного классификатора

Список литературы

- [Биргер, 1978] Биргер И.А. Техническая диагностика. – М. : Машиностроение, 1978. – 240 с. (2-е изд.: М. : URSS, 2019)
- [Поляк и др., 1973] Поляк Б.Т., Цыпкин Я.З. Псевдоградиентные алгоритмы адаптации и обучения // Автоматика и телемеханика. 1973. №3.
- [Davis et al., 2006] Davis J., Goadrich M. The relationship between Precision-Recall and ROC curves / Proceedings of the 23rd international conference on Machine learning. – Pittsburgh, 2006.
- [Klyachkin et al., 2017] Klyachkin V.N., Kuvayskova Yu.E., Zhukov D.A. The use of aggregate classifiers in technical diagnostics, based on machine learning // CEUR Workshop Proceedings. Vol.1903. Data Science. Information Technology and Nanotechnology, 2017.
- [Krasheninnikov et al., 2017] Krasheninnikov, V. R. / Models updating for technical objects state forecasting / V. R. Krasheninnikov; V. N. Klyachkin; Yu. E. Kuvayskova // 3rd Russian-Pacific Conference on Computer Technology and Applications (RPC). IEEE Xplore, 2018.
- [Neykov et al., 2016] Neykov, M. On the Characterization of a Class of Fisher-Consistent Loss Functions and its Application to Boosting / M. Neykov, Jun S. Liu, Tianxi Cai // Journal of Machine Learning Research. 2016. №17(70).
- [Witten et al., 2005] Witten I.H., Frank E. Data mining: practical machine learning tools and techniques. 2nd ed. – San Francisco : Morgan Kaufmann Publishers, 2005.

УДК 004.852

**СРАВНИТЕЛЬНАЯ ОЦЕНКА МЕТОДОВ
МАШИННОГО ОБУЧЕНИЯ В ПРОГНОЗИРОВАНИИ
ОТДАЛЕННЫХ РЕЗУЛЬТАТОВ
АОРТОКОРОНАРНОГО ШУНТИРОВАНИЯ¹**

Б.И. Гельцер (*boris.geltser@vvsu.ru*)
Дальневосточный федеральный университет, Школа
биомедицины, Владивосток

Р.С. Грицив (*griciv.roman@mail.ru*)
Владивостокский государственный университет экономики и
сервиса, Владивосток

В.Ю. Рублев (*groxmer@gmail.com*)
Дальневосточный федеральный университет, Школа
биомедицины, Владивосток

К.И. Шахгельдян (*carina.shahgeldyan@vvsu.ru*)
Владивостокский государственный университет экономики и
сервиса, Владивосток

Аортокоронарное шунтирование (АКШ) является одним из самых распространенных видов хирургического лечения ишемической болезни сердца. Прогноз отдаленных результатов этой операции представляет большой интерес для клинической медицины. В работе представлены пять прогностических моделей длительности жизни после АКШ, построенные на основе линейной регрессии, случайного леса и искусственных нейронных сетей. Сравнение моделей выполнялось с помощью четырех метрик, характеризующих абсолютную и относительную ошибку прогноза. Показано преимущество многослойной нейронной сети, позволяющий обеспечить прогноз с точность 74,3%.

Ключевые слова: методы машинного обучения, метрики точности моделей, отдаленные результаты аортокоронарного шунтирования

¹ Работа выполнена при финансовой поддержке РФФИ (проект № 18-29-03131).

Введение

В настоящее время аортокоронарное шунтирование (АКШ) является одним из самых распространенных видов хирургического лечения ишемической болезни сердца (ИБС), вследствие которого восстанавливается коронарный кровоток, увеличивается продолжительность и качество жизни пациентов [Rao, 2017]. Оценка отдаленных результатов АКШ является предметом многих медицинских исследований. Накопленный клинический опыт свидетельствует о том, что АКШ не может предотвратить рецидивирующую или прогрессирующую ишемию миокарда в полном объеме. Доказано, в частности, что через 2–3 года после АКШ рецидив стенокардии развивается у 3,5–7,2 % больных, а к пятому году их число возрастает до 36% [Zhang et al., 2011]. Частота и сроки развития сердечно-сосудистых событий обусловлены клиническими особенностями ИБС до АКШ, операционными факторами риска, наличием или отсутствием постинфарктного кардиосклероза, распространенностью и локализацией стенозирующего поражения коронарных артерий, наличием и активностью коморбидной патологии и др. Именно поэтому прогнозирование продолжительности жизни этих больных представляет особый интерес для клинической практики.

В настоящее время для этих целей применяют различные методы машинного обучения, которые позволяют по результатам анализа большего числа факторов выделить предикторы и разработать на их основе прогностические модели. Так, например, модель на основе случайного леса для оценки риска госпитализации по поводу хронической сердечной недостаточности в первые 30 суток после АКШ представлена в работе [Zolfaghar et al., 2013]. На основе дерева принятия решений предложена модель оценки риска развития сердечной недостаточности в послеоперационном периоде АКШ [Wang et al., 2012]. Нейронные сети использовались для оценки необходимости повторной реваскуляризации в течение 2 суток после операции [Goto et al., 2019], а также длительной вентилиации легких после АКШ [Wise et al., 2017]. Оценка риска смерти в течение месяца после операции выполняется с помощью шкалы EUROSCORE II, построенной на основе логистической регрессии [Nashef et al. 2012]. Этот методы, а также регрессия Кокса ранее были использованы для прогнозирования отдаленной смертности после АКШ [Forte et al., 2017], [Karim et al. 2018].

Анализ научных источников показывает, что исследования в этой области сосредоточены, в основном, на предсказании осложнений в сроки от 1 до 3 месяцев после операции. При этом моделирование отдаленных

результатов остается недостаточно изученным и актуальным.

Целью исследования является разработка и сравнительная оценка моделей на основе искусственного интеллекта для прогнозирования отдаленных результатов АКШ на примере продолжительности жизни больных после операции.

1. Материалы и методы

Для анализа были доступны данные 142 пациентов, оперированных в 2008–2018 гг. по поводу ИБС методом АКШ и умерших в сроки от 30 суток до 10 лет после операции. В исследование были включены 83 фактора, к которым относились возраст, пол, индекс курильщика, коморбидная патология, данные об особенностях оперативного вмешательства, включая осложнения, а также результаты клинико-лабораторных и инструментальных исследований. Из анализа были исключены пациенты, умершие от болезней, не связанных с ИБС. Результирующей зависимой переменной был фактор длительности жизни больных после АКШ, рассчитанный в месяцах. Таким образом, выходная зависимая непрерывная переменная находилась в диапазоне от 1 до 120 месяцев. Входные переменные – подмножество предикторов из исследуемых факторов, представляли собой непрерывные или категориальные переменные. Последние кодировались бинарными переменными, которые характеризовали присутствие у объекта соответствующей категории. Непрерывные переменные нормировались таким образом, чтобы их дисперсия была равна 1.

Для разработки прогностических моделей использовали три метода: линейный регрессионный анализ, метод случайного леса [Breiman, 2001] и искусственные нейронные сети [Flash, 2012]. При выборе линейных регрессионных моделей учитывали их значимость, а также значимость каждого предиктора (*p-value*). В качестве достоверных выбирали только те модели, которые имели *p-value* меньше 0,05. При построении линейных регрессионных моделей учитывали фактор мультиколлинеарности и не включали коррелированные переменные в одну модель. Для моделей на основе случайного леса и искусственных нейронных сетей фактор мультиколлинеарности не учитывали. Выбор моделей для дальнейшей оценки осуществлялся в этих случаях только с использованием метрик ошибок, описанных в соотношениях (1.1) – (1.4). Наличие нескольких моделей, построенных одним методом, обусловлено использованием в них разных предикторов, что связано, в свою очередь, с некоторой разреженностью исходных данных.

Для оценки качества моделей использовали несколько метрик.

Средняя абсолютная ошибка (Mean Absolute Error – MAE) позволяет оценить отклонение прогнозируемой длительности жизни после АКШ от реальной и вычисляется по формуле:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - z_i|, \quad (1.1)$$

где z_i – истинные значения длительности жизни пациента после операции в месяцах, а y_i – результат его моделирования. Вторая метрика – коэффициент детерминации (R^2), который вычисляли по формуле:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - z_i)^2}{\sum_{i=1}^n (z_i - \bar{z})^2}, \quad \bar{z} = \frac{1}{n} \sum_{i=1}^n z_i. \quad (1.2)$$

Третья метрика – это средний процент отклонения (Mean Absolute Percentage Error – MAPE), рассчитываемый по формуле:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - z_i|}{|z_i|} \quad (1.3)$$

И, наконец, для снижения значимости ошибок при малых значениях использовали метрику, полученную нормированием MAE:

$$PMAE = \frac{\sum_{i=1}^n |y_i - z_i|}{\sum_{i=1}^n |z_i|} = \frac{MAE}{|\bar{z}|}. \quad (1.4)$$

Данные о больных были разделены на 2 выборки: обучающую ($3/4$) и тестовую ($1/4$). Модели были разработаны на обучающей выборке и затем верифицированы на тестовой.

Все вычисления выполнялись с помощью языка R, в среде R-studio, а также пакетов randomForest [Breiman, 2001] и neuralnet [Günter et al., 2010].

2. Результаты

2.1. Линейные регрессионные модели длительность жизни после аортокоронарного шунтирования

На первом шаге разработки моделей в качестве наиболее вероятного предиктора, влияющего на продолжительность жизни после АКШ, был выбран возраст больных. Построенная линейная регрессионная модель на основе возраста имела высокую значимость, но низкие метрики точности. Далее ее совершенствовали ее путем добавления новых факторов. В тех случаях, когда добавленный фактор не обладал достоверностью, его исключали из модели. Если включение одного фактора приводило к снижению значимости других (что могло быть следствием мультиколлинеарности), то пересматривали информативность всей группы факторов и выбирали те из них, которые имели наилучшую значимость и обеспечивали максимальный уровень R^2 .

Первая регрессионная модель (PM1) позволяет рассчитать

длительность жизни после АКШ по следующему соотношению (2.1):

$$y=104,93-0,73 \cdot x_1-13,3 \cdot x_2-0,07 \cdot x_3-8,15 \cdot x_4+0,48 \cdot x_5, \quad (2.1)$$

где y – продолжительность жизни пациента после АКШ, x_1 – возраст пациента; x_2 - функциональный класс стенокардии (ФКС); x_3 - продолжительность операции в минутах; x_4 - количество наложенных шунтов; x_5 - фракция выброса крови до операции. Значимость модели составляла $p\text{-value}=3,32e-08$, $R^2 = 0,33$. Наибольшее влияние на продолжительность жизни в модели (2.1) оказывали фракция выброса крови, количество наложенных шунтов и ФКС.

Вторая регрессионная модель (PM2), включала дополнительные предикторы, которые не были использованы в PM1 (2.1) ввиду их отсутствия у части пациентов.

$$y=138,55-1,13 \cdot x_1-22,84 \cdot x_2-4,96 \cdot x_3+0,87 \cdot x_4-48 \cdot x_5, \quad (2.2)$$

где y – продолжительность жизни пациента после АКШ; x_1 – возраст; x_2 - ФКС; x_3 - содержание эозинофилов в крови до операции; x_4 - фракция выброса до операции; x_5 - нормализованное протромбиновое время. Значимость модели была $p\text{-value}=1,04e-06$. $R^2 = 0,58$. Наибольшее влияние на продолжительность жизни, согласно модели (2.2) оказывали ФКС, протромбиновое время и содержание эозинофилов в крови.

Для наглядности представления точности прогноза рассмотрели упорядоченные по длительности жизни фактические данные и соответствующие им результаты прогноза (рис.1).

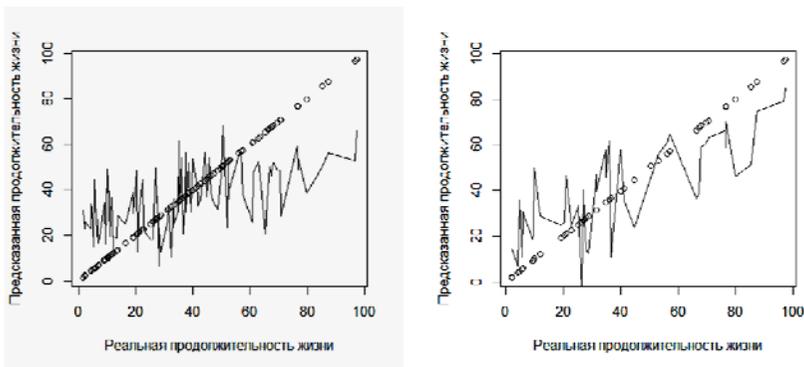


Рис. 1. График прогноза на основе линейных регрессионных моделей и фактических данных длительности жизни после АКШ в месяцах
а) PM1; б) PM2

Модель PM1 дает значительное расхождение с фактическими данными на горизонте продолжительности жизни после операции более 5 лет.

Прогноз по модели PM2 намного больше соответствует реальным данным.

2.2. Случайный лес для оценки длительности жизни после аортокоронарного шунтирования

Прогнозирование с помощью случайного леса (СЛ) популярно в медицине в виду того, что он не связан с проблемой переобучения, которая часто фиксируется в исследованиях на небольших выборках, что характерно для клинической медицины.

В модели СЛ для прогнозирования длительности жизни после АКШ использовалось большее число предикторов, чем в линейных регрессионных моделях. Наиболее значимыми из них были тромбиновое время и фракция выброса крови, при этом первый фактор был в 2 раза информативнее второго. Намного меньшую значимость имели показатели давления в легочной артерии, длительности операции, уровня гликемии, содержания миелоцитов и эритроцитов в крови, а также возраст пациентов. Коэффициент детерминации R^2 был равен 0,78, что свидетельствовало о значительной доле дисперсии длительности жизни после АКШ, объясненной построенной моделью. Соответствие прогноза реальным данным представлено на рис.2. Результаты моделирования с помощью СЛ отличались значительной ошибкой в первые 5 лет после операции и снижением ее уровня в более отдаленный период.

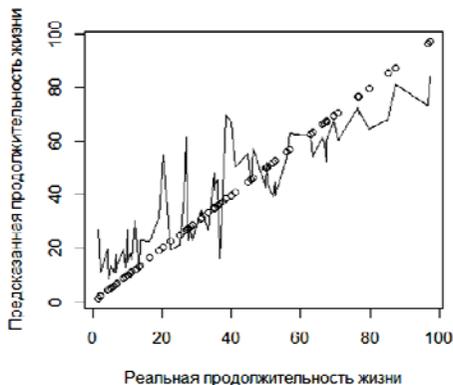


Рис. 2. График прогноза на основе модели СЛ и фактических данных длительности жизни после АКШ, месяцы

2.3. Искусственные нейронные сети для оценки длительности жизни после аортокоронарного шунтирования

Для построения модели длительности жизни использовали многослойную нейронную сеть с логистической функцией активации. Подбор архитектуры сети (количество слоев и нейронов в каждом слое), а также выбор предикторов выполняли, минимизируя *MAPE*. При этом, для принятия решения о выборе фактора в качестве предиктора или об архитектуре сети использовали до 1 млн. циклов обучения. Если после такой процедуры обучения метрика *MAPE* не уменьшалась, то выбранную архитектуру или введенный фактор корректировали.

В качестве предикторов для моделей на основе первой нейронной сети (ИНС1) выступали: возраст, количество шунтов и продолжительность операции, уровень гликемии, содержание лимфоцитов, эозинофилов и эритроцитов до операции и миелоциты - после, а также индекс курильщика. Во второй модели (ИНС2) использовали предикторы ИНС1, которые были дополнены следующими факторами: ФКС, давление в легочной артерии, а также показателем конечного систолического размера (КСР) по данным ЭхоКГ (рис.3). В обеих моделях R^2 был выше 0,9.

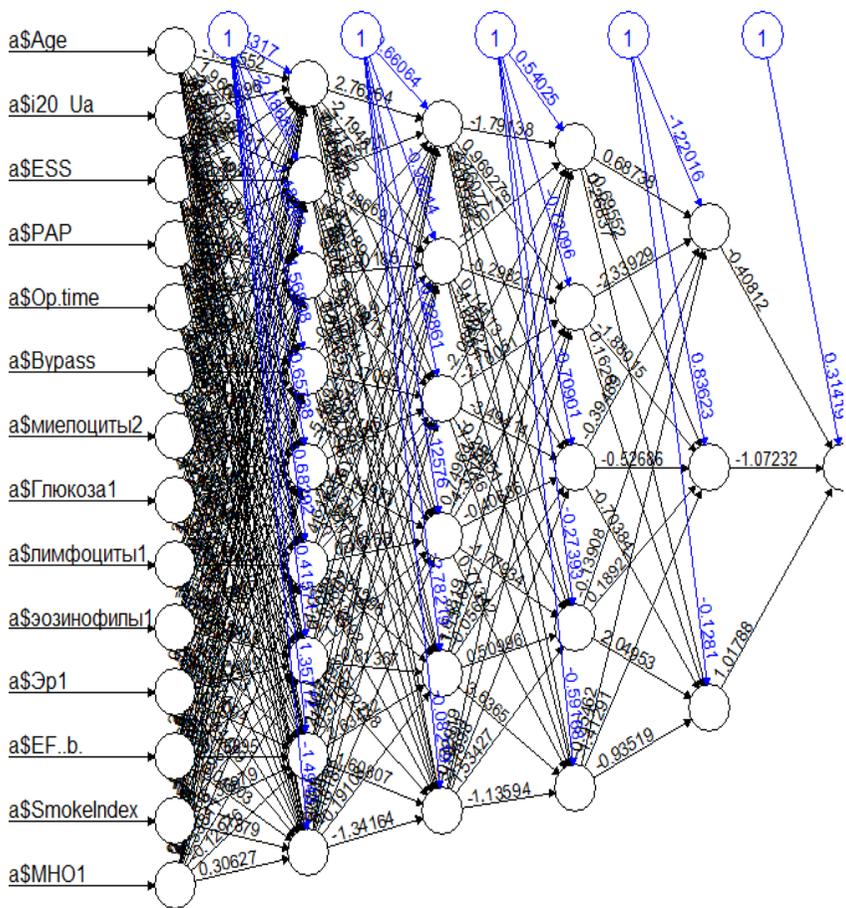


Рис.3. Архитектура ИНС2

Для оценки зависимости точности модели от ее величины построили график сопоставления фактической и прогнозируемой продолжительности жизни на основе двух моделей: ИНС1 и ИНС2 (рис.4).

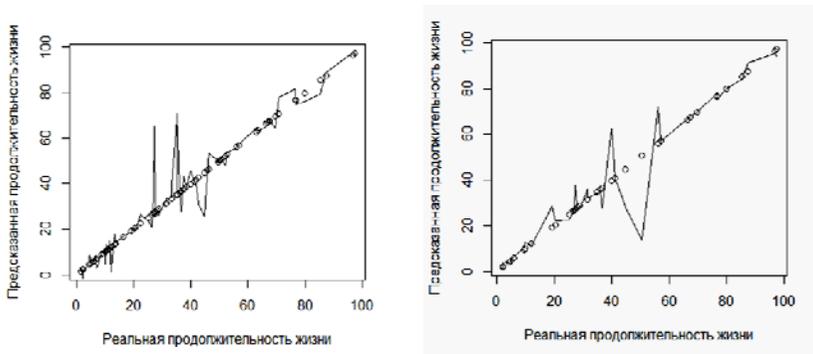


Рис.4. График прогноза на основе искусственных нейронных сетей и фактических данных продолжительности жизни после АКШ, месяцы
 а) ИНС1; б) ИНС2

3 Обсуждение

Для всех пяти построенных моделей рассчитаны метрики, характеризующие их качество (табл.1). Наилучшую точность показала модель ИНС2, которая прогнозировала продолжительность жизни после операции с точностью более 74% на тестовой выборке. Точность моделей, построенных с помощью линейной регрессии и случайного леса была ниже в среднем в 2,5 раза. Важно отметить, что метрика R^2 была недостаточно информативной для оценки качества моделей. Так, ее высокое значение (0,78) в модели СЛ не соответствовало низкой (55,5%) точности прогноза.

Табл.1

Точность моделей

Моде ли	R^2	Обучающая выборка			Тестовая выборка		
		<i>MAE</i>	<i>MAPE</i> , %	<i>PMAE</i> , %	<i>MAE</i>	<i>MAPE</i> , %	<i>PMAE</i> , %
PM1	0,33	16.2	129	44.3	11.9	64.3	43
PM2	0,58	15	94.7	36.7	17	56.4	54.1
СЛ	0,78	8.27	99	22	13.27	44.5	40.5

ИНС1	0,90	1,76	14,5	4,71	10,16	37,4	35,3
ИНС2	0,91	1,71	11,66	3,96	10,93	25,7	29,17

Набор предикторов для всех разработанных моделей приведен в табл. 2. Общими предикторами всех моделей были возраст и ФКС. Фракция выброса, количество шунтов, длительность операции и содержание в крови эозинофилов применяли в качестве предикторов в четырех моделях. В трех - использовали результаты лабораторных исследований: уровень гликемии, содержание эритроцитов и лимфоцитов в крови, а также индекс курильщика. Некоторые предикторы были задействованы только в одной или двух моделях: протромбиновое время, давление в легочной артерии, содержание миелоцитов и КСР. В научной литературе имеются единичные данные по долгосрочным прогнозам результатов АКШ. Так, в работе [Forte et al., 2017] строился прогноз пятилетней выживаемости после операции, где результирующим фактором была дихотомическая переменная, а в работе [Karim et al. 2018] был дан только трехлетний прогноз. Нами впервые используется непрерывная переменная в качестве результирующего фактора, а срок прогноза возрастает 10 лет.

Таблица 2

Предикторы моделей

	PM1	PM2	СЛ	ИНС1	ИНС2
Возраст	+	+	+	+	+
Индекс курильщика	-	-	+	+	+
ФКС	+	+	+	+	+
Длительность АКШ	+	-	+	+	+
Количество шунтов	+	-	+	+	+
Фракция выброса до операции	+	+	+	+	+
Давление в легочной артерии	-	-	-	+	+

Эритроциты	-	-	+	+	+
Эозинофилы	-	+	+	+	+
Лимфоциты	-	-	+	+	+
Миелоциты	-	-	+	+	+
КСР	-	-	-	+	+
Тромбиновое время	-	-	+	-	-
Нормализованное протромбиновое время	-	+	-	-	+
Уровень гликемии	-	-	+	+	+

Заключение

Авторами разработаны модели прогнозирования длительности жизни после АКШ на основе методов машинного обучения: регрессионных моделей, случайного леса и искусственных нейронных сетей. Лучшую точность моделирования отдаленных результатов АКШ обеспечили искусственные нейронные сети. Модель ИНС2 позволяет прогнозировать продолжительность жизни больных после АКШ в непрерывной шкале с горизонтом в 10 лет и точностью 74%. Методы машинного обучения также позволили выявить наиболее значимые предикторы длительности жизни.

Несмотря на то, что линейные регрессионные модели имеют меньшую точность по сравнению с СЛ и ИНС, они лучше интерпретируются, а патофизиологическое значение предикторов соответствует современным представлениям о патогенезе ИБС и сердечно-сосудистом риске после АКШ. На продолжительность жизни отрицательно влияют возраст, ФКС, содержание эозинофилов в крови, нормированное протромбиновое время, продолжительность операции и количество шунтов, а положительно – фракция выброса крови. Как известно, интерпретация ИНС является более сложной, однако, в нашем исследовании, используемые для их моделирования предикторы относятся в основном к известным индикаторам функционального статуса больных. Но были выявлены и

ранее не используемые, например, миелоциты, что является темой для дальнейшего исследования.

Список литературы

- [Breiman, 2001] Breiman L. Random Forests. // Machine Learning. 2001. № 45(1).
- [Flach, 2012] Flach P. Machine Learning. The Art and Science of Algorithms that Make Sense of Data. Cambridge. Cambridge University Press. (2012).
- [Forte et al., 2017] Forte J.C., Wiering M.A., Bouma H.R., Geus F., Epema A.H. Predicting long-term mortality with first week post-operative data after Coronary Artery Bypass Grafting using Machine Learning models // Proceedings of Machine Learning for Healthcare. 2017. USA
- [Goto et al., 2019] Goto S., Kimura M., Katsumata Y., Goto S., Kamatani T., Susaki J., Ichihara G., Ko Fukuda K., Sano M. Artificial intelligence to predict needs for urgent revascularization from 12-leads electrocardiography in emergency patients // PLOS. 2019. 14(1).
- [Günter, 2010] Günter F., Fritsch S. neuralnet: Training of Neural Networks. // The R journal. 2010. № 2(1).
- [Karim, 2018] Karim, M.N., Reid, C.M., Huq, M., et al. Predicting long-term survival after coronary artery bypass graft surgery. Interactive Cardiovascular Thoracic Surgery 2018. 26(2).
- [Rao, 2017] Rao M.P., Al-Khatib S.M., Pokorney S.D., et al. Sudden Cardiac Death in Patients With Ischemic Heart Failure Undergoing Coronary Artery Bypass Grafting: Results From the STICH Randomized Clinical Trial (Surgical Treatment for Ischemic Heart Failure). // Circulation. 2017. 135 (12).
- [Nashef et al. 2012] Nashef S.A.M., Roques F., Sharples L.D., Johan Nilsson, Christopher Smith, Antony R. Goldstone, Ulf Lockowandt. EuroSCORE II // European Journal of Cardio-Thoracic Surgery, 2012. 41 (4).
- [Wang et al., 2012] Wang Y., Simon M.A, Bonde P., Harris B.U., Teuteberg J.J., Kormos R.L., and Antaki J.F. Decision tree for adjuvant right ventricular support in patients receiving a left ventricular assist device. // Journal of Heart and Lung Transplantation. 2012. №31.
- [Wise et al., 2017] Wise ES, Stonko DP, Glaser ZA, et al. Prediction of Prolonged Ventilation after Coronary Artery Bypass Grafting: Data from an Artificial Neural Network. // Heart Surg Forum. 2017. 20(1):.
- [Zhang et al., 2011] Zhang X, Wu Z, Peng X. et al. Prognosis of diabetic patients undergoing coronary artery bypass surgery compared with nondiabetics: a systematic review and meta-analysis. // Journal of cardiothoracic and vascular anesthesia. 2011. №25.
- [Zolfaghar et al., 2013] Zolfaghar K., Meadem N., Teredesai A., Roy S.B., Chin SC. Big Data Solutions for Predicting Risk-of-Readmission for Congestive Heart Failure // Proc. IEEE International Conference on Big Data, USA. 2013.

УДК 004.825

МОДИФИКАЦИЯ АЛГОРИТМА СОПРЯЖЕНИЯ КОМПОНЕНТОВ ОТКРЫТЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Д.В. Жевнерчук (*zhevnerchuk@yandex.ru*)

ФГБОУ ВО Нижегородский государственный технический
университет им. Р.Е. Алексева, Нижний Новгород

В работе описывается система элементов и базовых отображений, позволяющая формализовать компоненты открытых информационных систем различного назначения в виде совокупности режимов сопряжения, точек доступа, стандартизированных свойств и доменов. Обсуждаются вопросы применения алгебраических типов данных и решеток для оптимизации алгоритма сопряжения компонентов.

Ключевые слова: открытая информационная система, алгебраические типы данных, алгебраическая решетка, онтология.

Введение

Создание инструментария интеллектуальной поддержки структурно-параметрического синтеза открытых информационных систем (ОИС) является одной из актуальных задач, носящих одновременно как фундаментальный, так и прикладной характер. Фундаментальность определяется необходимостью выявления системы элементов и преобразований, на основе которых строятся обобщенные и прикладные алгоритмы сборки целевых ОИС. Открытые информационные системы, как класс, могут быть представлены в виде связанных друг с другом компонентов, взаимодействующих через стандартизированные интерфейсы [Сухомлин, 1996], и процесс формирования многокомпонентных структур сводится к поиску пар компонентов, между которыми существует канал данных, не нарушающий режимы их функционирования и сопряжения. Наблюдаемая в последнее время тенденция, связанная с усложнением задач, относящихся к объектам

промышленности, крупных корпораций и к сферам государственной деятельности, приводит к необходимости повышения степени детализации при описании компонентной базы ОИС, что требует создания новых методов, обеспечивающих массовость, масштабируемость алгоритмов компонентной сборки ОИС различного назначения.

Классические комбинаторно-логические методы [Головицына, 2013] обеспечивают синтез многокомпонентных систем за счет информационного поиска целевой структуры из известного числа альтернативных фиксированных наборов компонентов, связи между которыми задаются вручную экспертом. Кроме того, тут не регламентированы правила описания интерфейсов, требований и ограничений, предъявляемых к ОИС.

Современные подходы к синтезу открытых информационных систем, как правило, ориентированы на частные случаи. Применение эволюционного моделирования [Курейчик и др., 2012] связано с необходимостью выражения структурных и функциональных аспектов ОИС в терминах биологических систем. Использование онтологического подхода [Курейчик и др., 2015] без введения системы базовых концептов и ролей приводит к семантически сложным моделям, большому числу связанных аксиом, требующим верификации. Также, в процессе онтологического описания ОИС требуется учитывать ограничения дескрипционных логик.

В работах [Жевнерчук, 2018], [Жевнерчук, 2019] предложен метод синтеза многокомпонентных интероперабельных структур (МИС), основанный на представлении стандартизированных компонентов, их интерфейсов, а также МИС в виде иерархии концептуальных каркасов онтологий, которая математически обоснована на уровне теоретико-множественного представления. Метод носит обобщенный характер, поэтому вопросы реализации и оптимизации конкретных алгоритмов компонентной сборки на базе предложенного метода требуют решений.

1. Применение алгебраических типов данных и решеток для формализации режимов и правил сопряжения элементов ОИС

Согласно работе [Гуляев и др., 2012], в качестве базовых элементов ОИС принято использовать подсистемы/сервисы/компоненты/модули, представляющие собой преобразователи информации, обладающие входными и выходными интерфейсами, которые должны быть стандартизированы и поддерживаться системой эталонных и прикладных спецификаций. Не зависимо от типа элемента, они могут быть

представлены абстрактными блоками, обладающими способностью принимать и передавать вонне потоки данных, при этом передаваемые данные должны определяться стандартизированными наборами

- Параметров P ;
- Доменов D_P , определяющих области допустимых значений, которые могут принимать параметры.

Для описания направления потоков данных, введем понятие точки доступа AP (access point) элемента ОИС, под которым будем понимать пару $(type, D_P)$, где $type$ – тип точки доступа, принимающий одно из двух значений: вход, выход. Домен задается совокупностью проекций базового типа данных и ограничений на нём. Режимы сопряжения элементов ОИС представлены подмножествами точек доступа, которые должны быть задействованы одновременно при формировании канала передачи данных между блоками.

На рис. 1 представлена схема элемента ОИС с указанием всех базовых множеств и отображений.

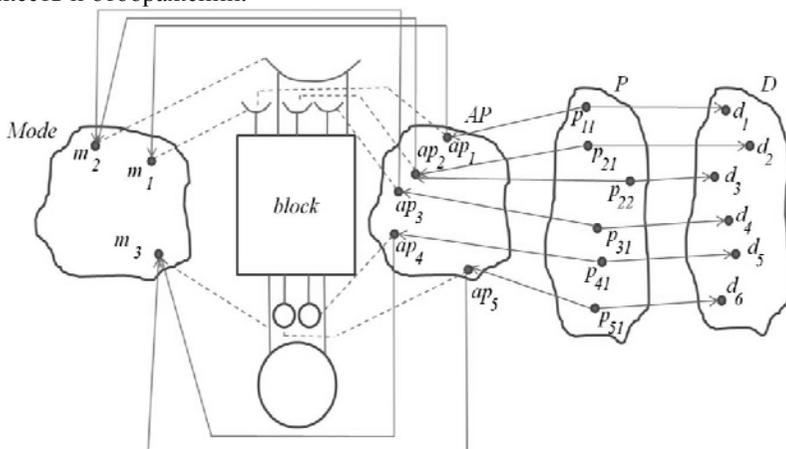


Рис.1. Связь элементов ОИС с доменами, свойствами, точками доступа и режимами сопряжения

Представленный элемент обладает тремя точками доступа типа «вход» ap_1, ap_2, ap_3 и двумя – типа «выход» ap_4, ap_5 , с которыми связаны свойства p_{ij} , где i – индекс точки доступа, j – индекс свойства потока данных, проходящего через точку доступа и домены d_k . С точкой доступа могут быть связаны один и более доменов, например, через ap_2 проходит поток данных, структура которого определяется двумя параметрами p_{21}, p_{22} , область допустимых значений которых задана доменами d_2 и d_3

соответственно. Элемент может работать в трех режимах сопряжения:

- В режиме m_1 блок работает как приемник данных в формате, задаваемом доменом d_1 через точку доступа ap_1 ;
- В режиме m_2 блок работает как приемник данных, причем в этом случае одновременно задействованы две точки входа ap_2, ap_3 , через которые поступает интегрированный поток данных в формате, задаваемом доменом $((d_2, d_3), d_1)$;
- В режиме m_3 блок работает как источник данных, причем в этом случае одновременно задействованы две точки выхода ap_4, ap_5 , через которые выдается интегрированный поток данных в формате, задаваемом доменом (d_5, d_6) ;

Очевидно, что точка доступа описывается произведением доменов, режим сопряжения может быть выражен произведением используемых точек доступа. При этом множество всех режимов сопряжения выражается суммой режимов сопряжения.

$$M = \bigcup_{i=1}^n \left(\prod_{j=1}^q \prod_{k=1}^w d_{jk} \right)_i \quad (1.1)$$

где d_{jk} – k -ый домен, связанный с j -ой точкой доступа, w – количество доменов, связанных с j -ой точкой доступа, q – количество точек доступа, определяющих i -ый режим сопряжения, n – количество режимов сопряжения элемента ОИС.

Таким образом, независимо от природы компонента, его режимы сопряжения определяются алгебраическим типом данных [Душкин, 2009], что, в свою очередь, позволяет обобщить понятие режима сопряжения и интерфейса и строить абстрактные модели многокомпонентных структур независимо от природы входящих в их состав элементов.

Два компонента считаются совместимыми, если между ними возможна передача данных через точки доступа, при этом домен, связанный с точкой доступа типа «вход», должен являться подмножеством домена, связанного с точкой доступа типа «выход». Домен может быть представлен в виде проекций на множество элементов базового типа данных, на множество ограничений, таких как, например, *enum* (перечисление), *range* (диапазон).

В общем случае, для проверки совместимости элементов ОИС, требуется выполнить попарное сравнение проекций доменов и в зависимости от базового типа, а также от контекста решаемой задачи, применить правила определения вложения одного множества допустимых значений в другое.

Чтобы не выполнять проверку вложенности при каждой сборке многокомпонентных структур, предлагается над множеством доменов

построить решетку, сделать множество доменов частично упорядоченным по операции « \subseteq ».

2. Онтологическое моделирование режимов сопряжения элементов ОИС

В работе [Жевнерчук, 2018] были предложен концептуальный каркас онтологии, который формализует открытую информационную систему на уровне доменов, интерфейсов и компонентов (рис. 2) и имеет следующие ограничения:

- Точка доступа элемента ОИС не идентифицируется, поскольку моделируется ролью *in* (для точек доступа типа «вход») и *out* (для точек доступа типа «выход»);
- Отсутствует возможность объединения точек доступа в режимы сопряжения элементов ОИС;
- С одной точкой доступа может быть связан только один домен;
- Два элемента ОИС могут сформировать двухкомпонентную структуру, если области допустимых значений, передаваемых через точки доступа типа «вход» и «выход» совпадают с точностью до базового типа данных и ограничений на этом типе;

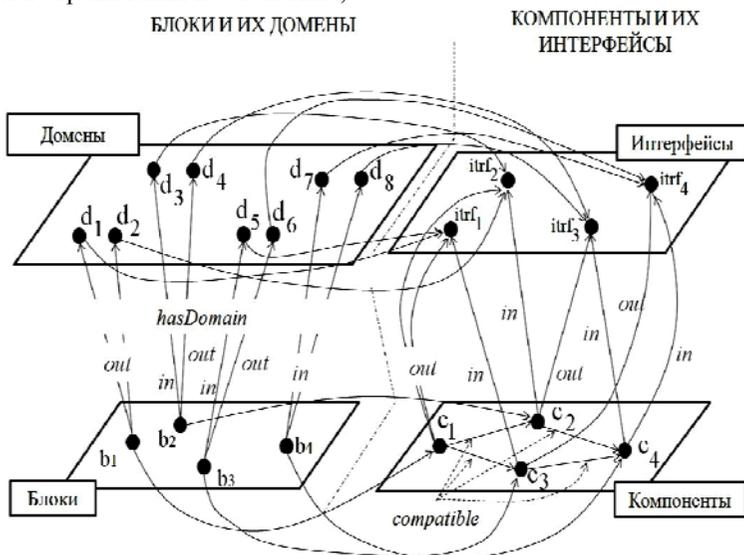


Рис.2. Концептуальный каркас онтологии доменов, интерфейсов и компонентов

Предлагается расширение онтологической модели описаниями:

- Режимов сопряжения компонентов, что позволит регистрировать и индексировать альтернативные варианты использования компонентов;
- Решетки доменов на основе операции вложения, что позволит индексировать вложение доменов друг в друга;

Введем концепт *Mode* («режим сопряжения»), индивиды которого будут связаны с индивидами блоков с помощью роли *hasMode* («поддерживает режим сопряжения»). Совокупность всех связей *hasMode* представляет собой множество всех режимов сопряжения элемента ОИС, при этом каждый из режимов связан ролью *includeAccessPoint* с точками доступа элемента, используемыми в выбранном режиме сопряжения.

Введем роль *isSubSet* («является подмножеством»), *isEqualTo* («равны»), в качестве объекта и субъекта выступают индивиды доменов.

Заключение

В работе предложена модификация алгоритмов сопряжения компонентов ОИС и формирования многокомпонентных структур на основе модернизированного онтологического каркаса, особенностью которой является индексирование каналов обмена данными и режимов сопряжения элементов ОИС. Таким образом, проверка вложенности доменов, определяющих возможность передачи данных между компонентами, выполняется один раз, при регистрации доменов, что уменьшает общее время работы алгоритма.

На следующем этапе предполагается создание параллельных версий алгоритмов сопряжения компонентов.

Список литературы

[Гуляев и др., 2012] Гуляев Ю.В., Журавлев Е.Е., Олейников А.Я. Методология стандартизации для обеспечения интероперабельности информационных систем широкого класса. Аналитический обзор // Журнал радиоэлектроники. 2012. №3. <http://elib.ict.nsc.ru/jspui/bitstream/ICT/1170/1/text.pdf>

[Головицына, 2013] Головицына М.В. Методы, модели и алгоритмы в автоматизированной подготовке и оперативном управлении производством РЭС. Москва: ИНФРА-М.2013.

[Kowalsky, 1975] Kowalsky R. A proof procedure using connection graphs. // Journal of the ACM. 1975. № 22(4).

[Курейчик и др., 2012] Курейчик В.В., Курейчик В. М., Родзин С.И. Теория эволюционных вычислений. Москва. ФИЗМАТЛИТ. 2012.

[Курейчик и др., 2015] Курейчик В.М., Курейчик Л.В. Построение онтологии для поиска нетривиальных знаний // Информатика, вычислительная техника и

инженерное образование, 2015. т.24. №4.

[**Сухомлин, 1996**] Сухомлин В.А. Методологический базис открытых систем // Открытые системы. <https://www.osp.ru/os/1996/04/178934>

[**Жевнерчук, 2018**] Жевнерчук Д.В. Концептуальные каркасы онтологий самоструктурируемых открытых информационных систем // Программная инженерия: методы и технологии разработки информационно-вычислительных систем. ПИИВС-2018. II международная научно-практическая конференция. – Донецк. 2018.

[**Жевнерчук, 2019**] Жевнерчук Д.В. Обобщенный метод синтеза многокомпонентных интероперабельных структур на основе онтологии и недетерминированного конечного автомата // Информационные технологии. – Москва: Новые технологии. 2019. том. 25. №2

[**Душкин, 2009**] Душкин Р. Алгебраические типы данных и их использование в программировании // Практика функционального программирования. 2009. вып. 2. <http://fprog.ru/2009/issue2/roman-dushkin-algebraic-data-types/>

УДК 004.89

ВОЗМОЖНОСТИ ПРИМЕНЕНИЯ ОНТОЛОГИЧЕСКОГО ПОДХОДА К ДИАГНОСТИКЕ ОРФАННЫХ ЗАБОЛЕВАНИЙ¹

Б.А. Кобринский (*kba_05@mail.ru*)
Н.А. Благосклонов (*nblagosklonov@gmail.com*)
Федеральный исследовательский центр «Информатика и
управление» РАН, Москва

Н.С. Демикова (*ns_d@rambler.ru*)
Российская Медицинская Академия Непрерывного
Профессионального Образования, Москва

В.В. Грибова (*gribova@iacp.dvo.ru*)
Е.А. Шалфеева (*shalfeeva@iacp.dvo.ru*)
М.В. Петряева (*margaret@iacp.dvo.ru*)
Институт автоматизации и процессов
управления ДВО РАН, Владивосток

Работа представляет первые результаты компьютерной диагностики редких заболеваний с использованием онтологической системы. Алгоритм включает обработку модальностей признаков с их коэффициентами в сочетании с комплексными характеристиками факторов уверенности признаков по возрастным группам пациентов. Это позволяет в дальнейшем получить узкий дифференциально-диагностический ряд заболеваний на долабораторном этапе обследования.

Ключевые слова: онтология, дифференциальная диагностика, редкие болезни, модальности признаков, факторы уверенности, интеллектуальная система поддержки принятия решений.

¹ Работа выполнена при частичной финансовой поддержке РФФИ (проект 17-07-00956).

Введение

К группе орфанных (редких) болезней в России принято относить заболевания, которые встречаются не чаще, чем 10 случаев на 100 000 населения (в соответствии с Федеральным законом от 21.11.2011 N323-ФЗ). В крупнейшей мировой базе генетически-обусловленных болезней Online Mendelian Inheritance in Man (OMIM) по состоянию на 26 марта 2019 года содержится информация о 6413 заболеваниях [OMIM, 2019]. Диагностика этой патологии вызывает у врачей серьезные трудности в виду необходимости учета разнообразных проявлений заболеваний на разных этапах ее развития. В последний период особенно актуальным является учет особенностей первых проявлений болезни (манифестации) для группы метаболических болезней накопления, где в качестве лечения применяется ферментозаместительная терапия и начаты экспериментальные работы по генной терапии [Das, 2013; Parini et al., 2010; Ellinwood et al., 2004]. Своевременно начатое лечение способно предупредить или затормозить дальнейшее прогрессирование патологии.

Для принятия диагностических решений необходимо учитывать множество факторов. Качество оказываемой медицинской помощи зависит уже не только от уровня подготовки (компетентности) врачебного персонала, но и от систем, осуществляющих поддержку решений врача [Purcell, 2005; A-Jasmi et al., 2010; Alves et al., 2016]. Используемые в настоящее время в клинической генетике системы [Baraitser et al., 2001; Köhler et al., 2009; POSSUM, 2019] не позволяют достаточно эффективно решать проблему диагностики многочисленных редких наследственных болезней в раннем возрасте.

Целью данной публикации является описание прототипа онтологической системы для поддержки принятия диагностических решений на модели наследственных лизосомальных болезней накопления.

Высокая степень сходства исследуемых болезней предполагает выдачу узкого дифференциально-диагностического ряда, позволяющего провести последующее направленное обследование, что важно в отношении ускорения распознавания заболевания и экономии средств дорогостоящих реактивов. Рассмотрение решения этой проблемы возможно с использованием методов искусственного интеллекта.

1. Представление знаний об орфанных заболеваниях

Извлечение знаний на 1 этапе осуществлялось из литературных источников и представлялось в форме текстологических карт [Kobriniskii et al., 2018]. В дальнейшем когнитолог с экспертами использовали их в

процессе обсуждения модальности и мер доверия к признакам. На 2 этапе эксперты формировали оценки факторов уверенности для манифестации признаков и их выраженности для разных возрастных групп.

В процессе построения прототипа экспертной системы для диагностики орфанных болезней на основе онтологической системы [Gribova et al., 2017] была выявлена целесообразность модификации атрибутов симптомов диагностируемых заболеваний и оценок факторов уверенности, описанных в [Кобринский, 2018]. Понятие наличия признаков было заменено понятием модальности (лат. *modus* — мера, оценка высказывания). Релевантные признаки были подразделены на главные, необходимые и второстепенные, что определяется их модальностью [GARD, 2019]. Признаки не только были распределены по модальности, что в определенной степени учитывало частоту их встречаемости, но и сопровождалось коэффициентами уверенности. Кроме того, были внесены изменения в шкалы мер доверия экспертов к манифестации и выраженности признаков, в большей степени учитывающие способы их определения врачами, что отражено в разделе 2 данной статьи.

2. Шкалы и формулы для оценки характеристик наследственных метаболических болезней

Формирование шкал модальности (M) признаков с их коэффициентами и формул для комплексной оценки совокупности факторов уверенности для лизосомальных болезней являлось необходимым условием при построении диагностических правил нечетко проявляющихся орфанных заболеваний.

Подбор количественных характеристик модальности подвергался изменениям в процессе анализа интегральных оценок. В результате модальность признаков была представлена градациями 5, 4 и 2, соответствующими лингвистическим характеристикам – главные, необходимые, второстепенные.

Манифестация (m) признака характеризовалась мерой доверия (уверенности) экспертов в том, что данный признак определённого заболевания манифестирует в конкретном возрасте (возрастной группе). Определяется в интервале $[-1; 1]$, где:

- -1 характеризует невозможность проявления данного признака в данном возрасте по физиологическим причинам, либо реализация признака завершилась в предыдущих возрастных группах;
- 0 характеризует норму, т.е. признак чисто теоретически может быть в данном возрасте, но либо не характерен для данного заболевания,

либо ещё не успевает проявиться в силу особенностей течения заболевания;

- $[0,1; 1]$ характеризует меру доверия экспертов к тому, что признак проявится (манифестирует) в одном из возрастных периодов.

Особенностью m является то, что значения, отличные от -1, в сумме не могут превышать 1. В каждом возрастном периоде указывается уверенность того, что признак манифестирует (обнаруживается) именно в данном возрасте. Факторы уверенности в манифестации по предшествующим четырем возрастным периодам имеют значения m_1, m_2, m_3, m_4 , дополняют друг друга, за исключением отрицательных величин.

Выраженность (s) характеризует уверенность экспертов в том, что данный признак встречается в конкретной возрастной группе с определенной степенью проявления. Определяется в интервале $[0; 10]$.

Изменение выраженности по возрастным периодам косвенно указывает на скорость развития патологических проявлений. Для каждого возрастного периода имеем: s_1, s_2, s_3, s_4 .

Для получения комплексной количественной оценки признака было предложено использовать следующую формулу:

$$P_i = M_i \cdot m_i \cdot s_i$$

где: P_i – параметр (признак), M_i – модальность (modality) признака, m_i – фактор уверенности манифестации (manifestation), s_i – фактор уверенности выраженности (severity).

3. Основные характеристики онтологии диагностики орфанных заболеваний

Одним из современных подходов к созданию систем поддержки принятия врачебных решений является онтологический. Одной из онтологий медицинской диагностики является Онтология знаний об идентификации заболеваний [Грибова и др., 2018]. Основными ее особенностями являются:

А. Возможность формирования симптомокомплексов, отражающих как динамику патологического процесса, так и текущее состояние организма.

Б. Возможность формирования разных симптомокомплексов заболеваний с учетом категорий пациентов (пол, возраст, особые состояния, например, беременность и др.) и разных подходов к диагностике (наиболее щадящий, быстрый или экономически оправданный и т.п.).

В. Возможность использования синдромального подхода к диагностике.

Г. Наличие простых и составных признаков (составной – это набор характеристик с множеством значений), расширенных модальностями, экспертными оценками, факторами выраженности значения признака.

Д. Возможность проведения дифференциальной диагностики заболеваний и выбора соответствующих методов лечения с учетом этиологии, патогенеза, развития процесса во времени.

Е. Учет причинно-следственных связей изменений под влиянием внешних воздействий, оказываемых на организм на разных этапах заболевания.

Ж. Возможность задавать необходимые условия для каждого заболевания: факторы, без которых невозможно заболевание (например, пол, время года и др.).

Среди указанных особенностей онтологии имеются необходимые для орфанных заболеваний.

4. Принципы функционирования онтологической диагностической системы

Для диагностики орфанных заболеваний возможны две ситуации:

1. Пациент впервые с подозрением (с признаками некоторых нарушений) обращается к врачу и требуется предложить адекватные гипотезы о заболеваниях, которые можно заподозрить.

2. Пациент имеет предварительный диагноз и требуется на основе имеющихся признаков и/или их динамики убедиться в наличии именно этого заболевания либо сформировать дифференциальный ряд для последующего уточнения другими специалистами и направленного проведения специальных исследований.

В зависимости от потребности практики для программного сервиса диагностики может быть сформирована база знаний (БЗ) с симптомокомплексами, отражающими либо текущее состояние организма, представленное набором возрастных категорий, либо с симптомами, отражающими динамику патологического процесса, либо с симптомокомплексами обоих типов. Каждый признак имеет шкалу его оценки, состоящую из модальности его проявления и экспертной количественной характеристики в виде факторов уверенности. Онтология медицинской диагностики-IACPaaS дает возможность определить связь заключений с объективными данными в истории болезни.

Используемая коллективом (авторами) форма представления онтологий и знаний на их основе (иерархические семантические сети) и применяемый инструментарий IACPaaS [Gribova et al., 2017] позволяют коллективно создавать программные сервисы диагностики (интегрируя

решатели с БЗ) и эффективно модернизировать базы знаний. В то же время, поиски семантического подобия в онтологиях представляют собой полезный способ использования семантической структуры фенотипических аномалий человека при решении проблемы дифференциальной диагностики.

5. Первые результаты эксперимента по применению онтологической системы в предметной области наследственной патологии

Для анализа результатов диагностики были взяты реальные истории болезни пациентов из отечественных и зарубежных открытых источников с верифицированными диагнозами. Истории болезни вводились через сервис ввода истории болезни платформы IACPaas и затем подавались на вход сервису диагностики орфанных заболеваний. Результатом работы сервиса является как подробное объяснение соответствия признаков пациента знаниям о заболевании, так и соответствия признаков пациента знаниям о разных сходных заболеваниях.

Особый интерес представляет распознавание болезней при неизвестном диагнозе. Пример дифференциальной диагностики в таком случае продемонстрировал вполне приемлемые результаты для долабораторного этапа идентификации нозологической природы заболевания.

При запуске Сервиса Поиска гипотез для ребенка 5 лет с подозрением на лизосомальное заболевание [Gabielli et al., 2016] было получено семь гипотез болезней из группы мукополисахаридозов. В пользу диагноза IS (синдром Шейе) указывали 4 принципиально важных признака (грубые черты лица, помутнение роговицы, гепатомегалия, тугоподвижность крупных суставов) при отсутствии полезных для подтверждения диагноза сведений о 2х признаках (задержка роста и снижение слуха).

В пользу диагноза мукополисахаридоз, тип II (синдром Гурлер) система выявила 5 принципиально важных признаков (грубые черты лица, помутнение роговицы, тугоподвижность крупных суставов, гепатомегалия, спленомегалия), но для однозначного подтверждения требовалась информация еще о 3х признаках (короткая шея, макроглоссия, умственная отсталость) (рис.1).

Таким образом, в пользу диагноза Шейе имелись 4 принципиально важных признака, а в пользу синдрома Гурлер 5 существенных признаков, но диагноз синдрома Шейе подтвержден 75% признаков (6 из 8), а синдрома Гурлер – 45% (7 из 15). Остальные 5 нозологий из группы мукополисахаридозов менее вероятны. Такой вариант соответствует

имеющему место в практике и является удовлетворительным.

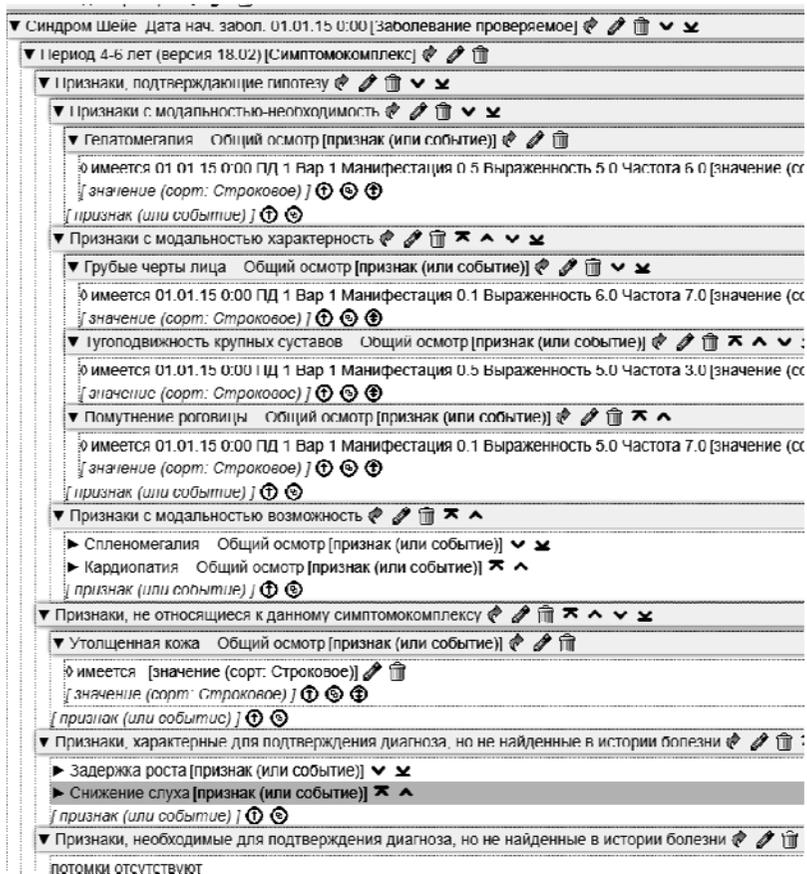


Рис.1. Фрагмент результата работы сервиса поиска гипотез при синдроме Шейе (без предварительной врачебной гипотезы).

6. Дискуссия

Семантические поиски сходства в онтологиях с использованием метрики осложняются тем фактом, что симптомы могут иметь различную степень специфичности, а также может иметь место наличие признаков, не связанных с данной болезнью [Köhler et al., 2009]. В зависимости от опыта врача и наличия специального оборудования (ЭХО-КГ, щелевая лампа и др.) клинические отклонения могут быть описаны более или

менее подробно. Для преодоления этого нами был разработан упомянутый выше комплекс из факторов уверенности с нестандартными интервалами оценок и коэффициенты модальности признаков. Также апробируются различные режимы работы онтологической системы.

Заключение

Высокая степень сходства исследуемых лизосомальных болезней накопления, что характерно и для большинства других наследственных орфанных болезней, предполагает получение не единственной возможной нозологической формы, а узкого дифференциально-диагностического ряда, позволяющего врачу провести последующее направленное лабораторное обследование, что важно в отношении ускорения распознавания заболеваний. Использование онтологической системы позволяет выдвинуть или подтвердить гипотезу врача с одновременным указанием на отсутствие необходимых для уточнения окончательного диагноза признаков.

Список литературы

[Грибова и др., 2018] Грибова В.В., Петряева М.В., Окунь Д.Б., Шалфеева Е.А. Онтология медицинской диагностики для интеллектуальных систем поддержки принятия решений // Онтология проектирования. 2018. Т.8. №1(27).

[Кобринский, 2018] Кобринский Б.А. Триединство факторов уверенности в задачах медицинской диагностики // Искусственный интеллект и принятие решений. 2018. №2.

[A-Jasmi et al., 2010] A-Jasmi F., Moldovan L., Clarke J.T. Hunter disease eClinic: interactive, computer-assisted, problem-based approach to independent learning about a rare genetic disease // BMC Med Educ. 2010. 10:72.

[Alves et al., 2016] Alves R., Piñol M., Vilaplana J., Teixidó I., Cruz J., Comas J., Vilaprinyo E., Sorribas A., Solsona F. Computer-assisted initial diagnosis of rare diseases // Peer J. 2016. 4.

[Baraitser et al., 2001] Baraitser M., Winter R.M. London Dysmorphology Database, London Neurogenetics Database & Dysmorphology Photo Library on CD-ROM. 3rd ed. – Oxford: Oxford University Press, 2001.

[Das, 2013] Das S.K. Inborn Errors of Metabolism: Challenges and Management // Indian J Clin Biochem. 2013. 28(4).

[Ellinwood et al., 2004] Ellinwood N.M., Vite C.H., Haskins M.E. Gene therapy for lysosomal storage diseases: the lessons and promise of animal models // J Gene Med. 2004. 6.

[Gabrielli et al., 2016] Gabrielli O., Clarke L.A., Ficcadenti A., Santoro L., Zampini L., Volpi N., Coppa G.V. 12 year follow up of enzyme-replacement therapy in two siblings with attenuated mucopolysaccharidosis I: the important role of early treatment // BMC Medical Genetics. 2016. 17:19.

[**GARD, 2019**] GARD. Genetic and Rare Diseases Information Center. <https://rarediseases.info.nih.gov/>, last accessed 2019/04/15

[**Gribova et al., 2017**] Gribova V., Kleshev A., Moskalenko P., Timchenko V., Fedorischev L., Shalfeeva E. The IACPaaS cloud platform: Features and perspectives // Computer Technology and Applications (RPC), 2017 Second Russia and Pacific Conference on. - IEEE, 2017.

[**Kobrinskii et al., 2018**] Kobrinskii B.A., Blagosklonov N.A. Hybrid approach to knowledge extraction: textual analysis and evaluations of experts // Open Semantic Technologies for Intelligent Systems. Материалы Международной научно-технической конференции. Вып.2. Республика Беларусь, Минск, 15-17 февраля 2018. – Минск: БГУИР, 2018.

[**Köhler et al., 2009**] Köhler S., Schulz M.H., Krawitz P., Bauer S., Dölken S., Ott C.E., Mundlos C., Horn D., Mundlos S., Robinson P.N. Clinical Diagnostics in Human Genetics with Semantic Similarity Searches in Ontologies // American Journal Human Genetics. 2009. 85(4).

[**OMIM, 2019**] Online Mendelian Inheritance in Man® An Online Catalog of Human Genes and Genetic Disorders. <https://www.omim.org/>, last accessed 2019/04/15

[**Parini et al., 2010**] Parini R., Andria G. Lysosomal storage diseases: early diagnosis and new treatments // Mariani Foundation paediatric neurology. Vol.23. – Chicago: John Libbey Eurotext, 2010.

[**POSSUM, 2019**] POSSUM. Pictures of Standard Syndromes and Undiagnosed Malformations. <https://www.possum.net.au/>, last accessed 2019/04/15.

[Purcel, 2005] **Purcell G. What makes a good clinical decision support system // BMJ. 2005.Vol.330.**

УДК 519.711.3

ОНТОЛОГИЯ ИНЖЕНЕРНЫХ СИСТЕМ ЗДАНИЙ ДЛЯ ЦИФРОВОГО СТРОИТЕЛЬСТВА

О.Ю. Марьясин (*maryasin2003@list.ru*)

Ярославский государственный технический университет,
Ярославль

Рассмотрено применение различных технологий интеграции и семантической паутины для обмена и доступа к информации в цифровом строительстве. Предложен вариант организации единой информационной среды для цифрового строительства и разработана онтология инженерных систем зданий. Представленные решения позволяют проводить энергомоделирование, моделирование инженерных систем зданий и доступ к данным моделей не только на этапе проектирования, но и на других этапах жизненного цикла зданий.

Ключевые слова: информационная модель здания, энергомодель, цифровое строительство, инженерные системы зданий, семантическая паутина, онтология

Введение

В последнее время при проектировании зданий все шире используются технологии информационного моделирования зданий (BIM – Building Information Modeling), которые составляют основу цифрового строительства (Digital Construction). Проблемы сохранения окружающей среды и снижения коммунальных платежей требуют выполнения мер по энергосбережению и повышению энергетической эффективности при проектировании зданий. В то же время необходимо гарантировать комфортную и здоровую среду в помещениях. Все это создает благоприятные условия для более широкого использования моделирования энергопотребления зданий (BEM – Building Energy Modeling). Энергомоделирование в России, как и во многих других странах, пока является добровольным делом, если только не нужно сертифицировать здание на соответствие “зеленым” стандартам, таким как GREEN ZOOM [GREEN ZOOM, 2019]. Однако, при проектировании систем отопления, вентиляции и кондиционирования (ОВК или HVAC – Heating, Ventilation, & Air Conditioning) все равно, в том или ином виде,

приходится выполнять расчеты, связанные с теплообменом, обеспечением качества воздуха в помещениях и энергосбережением.

Сложность строительной инфраструктуры, связанная с множеством различных типов зданий, высокие требования к исполнению и эксплуатации инженерных сетей зданий приводят к тому, что при их проектировании необходимо обеспечить междисциплинарное взаимодействие между различными системами здания для выполнения комплексных расчетов и моделирования. Данные о конструкции и материалах здания из BIM, должны использоваться для проведения энергомоделирования в BEM, а построенные там модели использоваться при проектировании HVAC систем. Данные о моделях и HVAC оборудовании используются при синтезе системы управления зданием, потом вместе с данными о системе управления при проектировании информационной сети здания и других систем. При этом, сейчас, большинство операций по передачи данных, как правило, осуществляются вручную и отсутствуют общепринятые форматы передачи информации между многими системами.

Доступ к данным важен не только на этапе проектирования здания, но и на других этапах жизненного цикла здания, таких как строительство, логистика, эксплуатация и реконструкция. При этом для повышения эффективности и удобства работы с информацией необходимо обеспечить, чтобы нужные данные были доступны постоянно и откуда угодно. Для этого необходимо организовать круглосуточный доступ к информации через Интернет.

Технологии семантической паутины (Semantic Web), связанных данных (Linked Data) или связанных открытых данных (Linked Open Data – LOD) [Semantic Web, 2019] позволяют решить указанные проблемы и являются одним из эффективных способов обеспечения интероперабельности между различными системами. При этом базовую роль в стеке технологий семантической паутины играют онтологии. Использование онтологий облегчает совместное использование информации между различными системами, между людьми и программами или между программными агентами, позволяет выполнить отделение знаний в предметной области от оперативных знаний, контролировать целостность и непротиворечивость данных. Онтологии, в отличие от простого обмена данными с помощью документов в формате XML, могут определять не только синтаксис, но и семантику передаваемых данных. Для кодирования онтологий применяются языки RDF и OWL.

В данной работе автор рассматривает применение различных технологий интеграции и семантической паутины для обмена и доступа к информации в цифровом строительстве. Основное внимание при этом уделяется разработке онтологии инженерных систем здания.

1. Технологии интеграции и семантической паутины в цифровом строительстве

RVML [Юрин, 2016] – язык визуального моделирования правил, предназначенный для моделирования и описания продукций. RVML основан на UML и может быть рассмотрен как его профиль расширения использующий терминологию диаграмм классов, поэтому в качестве основных элементов используются понятия «класс» и «ассоциация». Данный язык позволяет абстрагироваться от различных производственных ЯПЗ, представляя логические правила в обобщенном виде.

При этом уточнение отдельных элементов языка (например, приоритет правила (P), значение коэффициента уверенности (KУ), значение слота «по умолчанию» и др.) позволяет учесть особенности некоторых производственных ЯПЗ, например, CLIPS или FuzzyCLIPS.

Описание основных элементов RVML может быть представлено в расширенной нотации Бэкуса–Наура (РБНФ):

```

<RVML> = {<Fact>}, {<Template>}, {<Rule>}, {<Relationship>}.
<Rule> = (<Condition>{<Condition>}), <Core>, (<Action>{<Action>}).
<Condition> = <C_Element>.
<Action> = <A_Element> {<A_Element>}.
<C_Element>
    = (<C_Operator>,
      <Fact>{<Fact>})
  |
  (<C_Operator>, <C_Element>).
<A_Element> = <A_Operator>, (<Fact>{<Fact>}).
<A_Operator> = Add | Delete | Modify | Stop.
<C_Operator> = AND | OR | NOT.
<Fact> = <Name>, <CF>, <F_Slot> {<F_Slot>}.
<F_Slot> = <Name>, <Constraint>, <Value>.
<Core> = <Name>, <CF>, <P>.
<Name> = <String>.
<Value> = <Number> | (<Symbol> {<Symbol>}) | <Set>.
<Constraint> = > | < | = | >= | <= | <>.
<CF> = [0,1]. (the certainty factor)
<P> = [1,100]. (the priority of the rule)
<Template> = <Name>, <T_Slot> {<T_Slot>}.

```

<T_Slot> = <Name>, <DataType>, [<Constraint>, <Default_Value>].

<DataType> = String | Symbol | Number.

<Default_Value> = <Value>.

<String> = <Symbol> {<Symbol>}.

<Relationship> = <Element>, <Connection>, <Element>.

<Connection> = <Kind>, <Name>.

<Kind> = <Association> | <Realization>

<Element> = <Condition> | <Action> | <Template> | <C_Operator>

Графические примитивы RVML основаны на графическом примитиве UML «класс», в частности:

Шаблон факта используется для создания фактов и частей правила, таких как условие и действие (рис. 1).

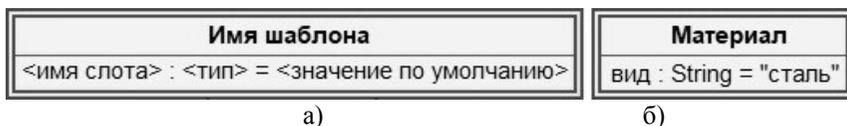


Рис. 1. Шаблон факта в RVML: а) общее представление; б) пример

Факт как элемент условия логического правила используется для представления фактов, над которыми проводятся манипуляции в рабочей памяти. Данный примитив позволяет представить не только описание факта (слоты и их значение), но и коэффициент уверенности (КУ) (рис. 2).



Рис. 2. Факт в RVML (как условие правила): а) общее представление; б) пример

Факт как действие в правиле представлена на рис.3.



Рис. 3. Факт в RVML (как действие правила): а) общее представление; б) пример

Логическое правило представляется в виде композиции экземпляров трех примитивов: фактов, как условий; фактов, как действий и «ядра» как узлового элемента для связи условий и действий. Узловой элемент позво-

ляет представить наименование правила, коэффициент уверенности (КУ) и приоритет (В) (рис. 4).

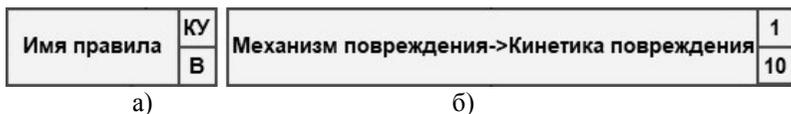


Рис. 4. Узловой элемент правила в RVML: а) общее представление; б) пример

Соединение между элементами (рис.5), представлена как ассоциация, содержащая узел для отображения операции с фактами в рабочей памяти:
 «+» – добавление; «-» – удаление; «~» – изменение; «!» – остановка логи- ческого вывода.

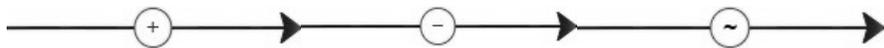


Рис. 5. Соединение между элементами в RVML

На рис. 6 представлен пример конструкций RVML: шаблон правила (обобщенное правило).



Рис. 6. Примеры конструкций RVML (шаблон правила)

RVML позволяет: использовать отдельные графические примитивы для отображения всех элементов продукции, а не стереотипы или типизированные классы как в UML; присваивать отдельным фактам субъективные вероятности в виде коэффициентов уверенности; наглядно отображать тип выполняемых действий, таких как добавление, изменение, удаление.

2. Обзор существующих онтологий

В настоящее время разработано уже достаточно много онтологий косвенно связанных с рассматриваемой предметной областью. Из них можно назвать онтологии из области “Умных домов” и автоматизированных систем управления зданиями (Building Management System – BMS), такие как SAREF, Brick, BOnSAI. Цель SAREF – охватить фундаментальные концепции в области интеллектуальных устройств [Daniele et al., 2015]. SAREF также включает некоторые классы для поддержки инженерных систем зданий. К ним относятся классы EnergyRelated, FunctionRelated, HVAC, Sensor, Meter, Actuator, Measurement, Property и другие. Однако набор доступных устройств, функций и команд в SAREF очень ограничен. Онтология Brick [Balaji et al., 2016] была разработана на основе SAREF. Она охватывает такие типовые подсистемы BMS, как система отопления, вентиляции и кондиционирования (Heating, Ventilation, & Air Conditioning – HVAC) и система освещения, пространственную и энергетическую инфраструктуру здания, а также отношения между этими объектами. Основные классы онтологии – Tag, TagSet, Equipment, Location, MeasurementProperty. Возможные отношения ограничены семью различными типами отношений для моделирования пространственных отношений, отношений композиции/агрегации и отношений ввода-вывода. Онтология BOnSAI [Stavropoulos et al., 1975] ориентирована на услуги и веб-сервисы в среде “Умного дома”. Разработанные основные понятия являются либо очень абстрактными, либо специфичными для веб-сервисов. В целом, набор доступных в онтологии типов устройств, функций, местоположений и параметров среды слишком ограничен для коммерческих систем автоматизации зданий.

В области “Умных домов” с акцентом на энергосбережение и системы управления энергопотреблением (Building Energy Management System – BEMS) можно выделить онтологии DogOnt и ThinkHome. В центре внимания онтологии DogOnt [Butzin, 2017] находятся системы “Умный дом” и вопросы моделирования энергопотребления устройств. Одним из недостатков DogOnt является ориентация понятия Location (расположение) на типы помещений (ванная комната, офис, гостиная), а не на пространственную планировку. Это затрудняет применение онтологии для описания крупных зданий. Кроме того, в ней не моделируется такая важная инженерная система как HVAC, имеющая большое значение в BMS. Онтология ThinkHome [Kofler, 2012] также

описывает энергосберегающие системы “Умный дом”. Она содержит понятия для описания комфорта, актора (пользователя), процесса, энергии, ресурса (устройства), внешних воздействий (погоды) и структуры здания.

В области базовых технологий контроля и мониторинга следует отметить онтологии SSN, M3, OntoSensor. Онтология SSN [SSN, 2019] предложена консорциумом W3C для описания предметной области, связанной со сбором данных с помощью датчиков и процессом наблюдения (сбора данных). Она основана на шаблоне проектирования Стимул – Датчик – Наблюдение и совместима со стандартами SensorML и O&M (Observations and Measurements) консорциума OGC (Open Geospatial Consortium). Поскольку онтология SSN предоставляет только основные понятия, ее необходимо расширять терминами, специфичными для конкретной предметной области. Онтология M3 [Gyraud, 2012] является развитием онтологии SSN для поддержки описания датчиков, наблюдений и их единиц измерения. Онтология OntoSensor [Xue, 2012] расширяет концепции, представленные стандартом SensorML для идентификации категорий датчиков, поведения, взаимосвязей, функций и метаданных, касающихся характеристик датчиков, их производительности и надежности.

3. Онтология инженерных систем зданий

Автор, на основе анализа существующих онтологий и модели данных системы EnergyPlus, разработал свою онтологию инженерных систем здания. К разрабатываемой онтологии предъявлялись следующие основные требования: она должна включать основные понятия и отношения предметной области; она должна быть ориентированной на систему энергомоделирования EnergyPlus; она должна быть модульной; она должна быть ориентированной на оптимизацию энергопотребления, современные методы автоматического управления, включая оптимальные и интеллектуальные системы. К сожалению ни одна из существующих онтологий, не удовлетворяет указанным требованиям и не подходит в качестве онтологии инженерных систем зданий.

Разработанная онтология построена путем объединения и выравнивания базовых онтологий. Набор базовых онтологий включает:

онтологию здания, включающую информацию о конструкции, внутренней и внешней среде здания, а также классы, относящиеся к конструкции и материалам здания, системы EnergyPlus;

онтологию инженерного оборудования, содержащую информацию об инженерном оборудовании здания и технических устройствах, а также классы EnergyPlus, относящиеся к HVAC оборудованию;

онтологию моделей, включающую классы, описывающие модели в системе EnergyPlus;

онтологию измерений, описывающую измеряемые параметры и процесс измерения;

онтологию данных, включающую различные источники и форматы данных, а также онтологию EXPRESS;

онтологию времени;

онтологию ресурсов, включающую информацию об основных энергоресурсах, их поставщиках и потребителях;

онтологию, включающую классы для обмена данными с EnergyPlus.

Из упомянутых онтологий только онтологии времени (<http://www.w3.org/2006/time#2016>) и онтология EXPRESS (<https://w3id.org/express>) являются стандартными онтологиями, остальные онтологии были разработаны автором. Одни из онтологий уже проработаны достаточно подробно, другие включают только основные понятия. Процесс формирования базовых онтологий еще не завершен.

Для разработки онтологии использовался популярный редактор онтологий Protégé. Фрагмент иерархии классов, разработанной онтологии в редакторе Protégé показан на рис. 7. Структура классов онтологии включает основные классы: Building, Data, Device, EPHVAC, Environment, Equipment, Event, Location, MeasurementProperty, Models, Point, Resource, RoomAirModels, State, SurfaceConstructionElements, TechnicalSystem, ThermalZonesAndSurfaces, ZoneHVAC и другие. Такая структура классов позволяет обеспечить компромисс между выразительной силой онтологии и эффективностью использования знаний, представленных онтологией. Рассмотрим некоторые основные классы онтологии более подробно.

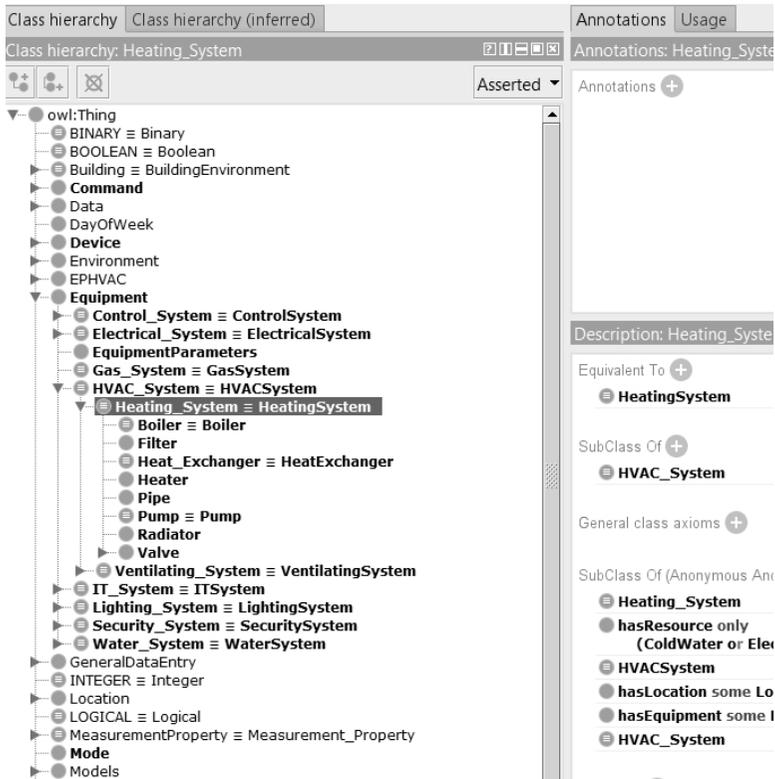


Рис. 7. Фрагмент иерархии классов в Protégé

Класс `Data` из онтологии данных содержит классы `DataSource`, `DataStorage` и `DataType`, предназначенные для хранения и представления информации. Класс `DataSource` включает классы `Document`, `File`, `Package`, `DataBase`, а класс `DataType` содержит описание основных типов данных. Онтология выровнена по основным типам данных `Binary`, `Boolean`, `Integer`, `Number`, `Real` и другим с онтологией EXPRESS.

Класс `Environment` онтологии здания представляет собой среду здания. Он включает классы: `IndoorEnvironment`, представляющий внутреннюю среду здания, `OutdoorEnvironment` представляющий внешнюю среду, `BuildingEnvironment`, включающий элементы конструкции здания (`BuildingElement`) и его внутреннее пространство (`BuildingSpace`), `EnvironmentalConditions`, включающий основные параметры внутренней среды здания, такие как температура, влажность, освещенность и концентрация CO_2 .

Класс Equipment онтологии инженерного оборудования включает классы инженерных систем здания: HVAC_System для системы отопления, вентиляции и кондиционирования (HVAC), Electrical_System для системы электроснабжения, Lighting_System для системы освещения, Water_System – системы водоснабжения, Gas_System – системы газоснабжения, Security_System – системы безопасности, Control_System – системы управления зданием, IT_System – информационной системы здания, а также класс EquipmentParameters для параметров инженерного оборудования. Онтограф для класса Equipment показан на рис. 8. Каждый класс инженерных систем включает классы основного инженерного оборудования соответствующих систем. Наиболее проработанным классом онтологии инженерного оборудования является класс HVAC_System, поскольку к базовым классам этого класса Boiler, Heat_Exchanger, Heater, Pipe, Pump, Radiator, Chiller, Coil, Fan было добавлено множество классов инженерного оборудования системы EnergyPlus.

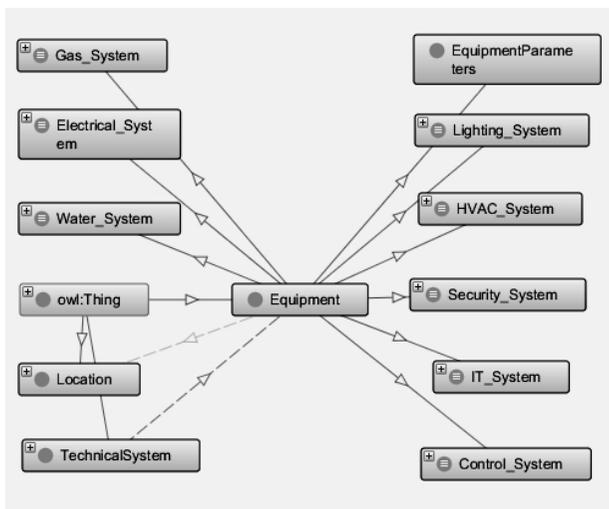


Рис.8. Онтограф для класса Equipment

Класс Location онтологии здания предназначен для описания местоположения. Он включает подклассы BuildingLocation эквивалентный классу BuildingSpace и Outside эквивалентный классу OutdoorEnvironment. Класс MeasurementProperty онтологию измерений представляет основные измеряемые физические свойства, используемые различными системами контроля, мониторинга и инженерным оборудованием. Он включает такие

параметры как температура, давление, расход, энтальпия, скорость, ток, напряжение, частота и многие другие. Класс Resource онтологии ресурсов представляет основные ресурсы здания. Он включает классы Resource Type для обозначения видов ресурсов, Resource Provider для поставщиков ресурсов, Resource Consumer для потребителей ресурсов и класс Resource Parameters для параметров ресурсов. Класс Resource Type в свою очередь включает такие виды ресурсов как, электроэнергия, тепло, холод, холодная и горячая вода, газ.

Для классов и их индивидуальностей в онтологии определено множество свойств (отношений). Часть их них представлена на рис. 4. Например, объекты класса Equipment онтологии инженерного оборудования связаны отношением hasLocation с объектами класса Location онтологии здания. Это отношение устанавливает связь между оборудованием и местом где оно установлено. Свойство hasLocation имеет обратное свойство isLocatedIn, связывающее объекты класса Location с объектами класса Equipment. В онтологии имеется много подобных инверсий свойств. Часть свойств унаследована от базовых онтологий. Другая часть устанавливает отношения между объектами из разных базовых онтологий.

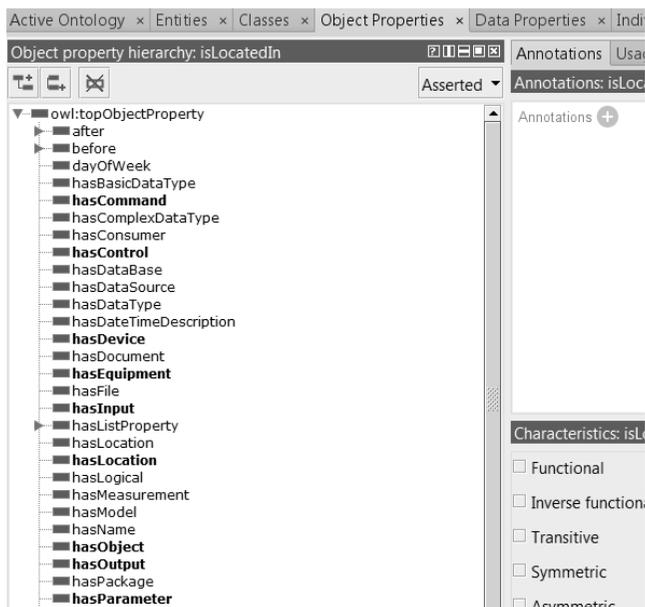


Рис.4. Отношения онтологии

Заключение

В работе рассмотрено применение различных технологий интеграции и семантической паутины для обмена и доступа к информации в цифровом строительстве. В результате работы предложен вариант организации единой информационной среды для цифрового строительства и разработана онтология инженерных систем здания.

Представленная архитектура единой информационной среды позволяет проводить энергомоделирование, моделирование инженерных систем зданий и доступ к данным моделей не только на этапе проектирования, но и на других этапах жизненного цикла зданий. Другим достоинством описанного варианта организации информационной среды является открытость и свободная-доступность ее основных компонентов, а, следовательно, низкая стоимость.

Особенностью разработанной онтологии является ее ориентация на решение задач организации обмена информацией между ВЕМ системой EnergyPlus и другими системами проектирования. Другой особенностью онтологии является ее высокая модульность. Практически при разработке данной онтологии был создан целый набор независимых онтологий, пригодных для повторного использования. К достоинствам онтологии по сравнению с подобными онтологиями, также можно отнести более проработанную и выразительную реализацию таких классов как Data, Environment, Equipment, Location, Model, Point, Resource, State.

Разработанная онтология была использована для проведения комплексного моделирования инженерных систем корпуса “Д” Ярославского государственного технического университета. С помощью онтологии и программного обеспечения на языке Python был организован процесс моделирования, доступ к данным моделей и обмен информацией между системой EnergyPlus и пакетами JModelica и MATLAB/Simulink. Однако возможности применения разработанной онтологии могут быть гораздо шире. Использование онтологии значительно облегчает как обмен данными между различными системами на стадии проектирования, так и доступ к информации на других стадиях жизненного цикла зданий.

Список литературы

- [Annex 60, 2019] IEA EBC Annex 60. – <http://www.iea-annex60.org/index.html>
- [Balaji et al., 2016] Balaji B., Bhattacharya A., Fierro G., Gao J., Gluck J., Hong D., Johansen A., Koh J., Ploennigs J., Agarwal Y., Berges M., Culler D., Gupta R., Kjergaard M. B., Srivastava M., Whitehouse K. Brick: Towards a unified metadata schema for buildings // Proc. 3rd ACM International Conference on Systems for Energy-Efficient Built Environments, USA. 2016.

[Butzin, 2017] Butzin B., Golatowski F., Timmermann D. A survey on information modeling and ontologies in building automation // Proc. 43rd Annual Conference of the IEEE, 2017

[Daniele et al., 2015] Daniele L., den Hartog F., Roes J. Study on Semantic Assets for Smart Appliances Interoperability, 2015.

[EnergyPlus, 2019] EnergyPlus. – <https://energyplus.net/>

[eplus-lod, 2019] GitHub-IMI-KIT/eplus-lod: Provides EnergyPlus I/O interoperability via semantic web technologies. – <https://github.com/IMI-KIT/eplus-lod>

[FMI, 2019] Functional Mock-up Interface. – <http://fmi-standard.org/>

[Fuseki, 2019] Apache Jena Fuseki. – <https://jena.apache.org/documentation/fuseki2/index.html>

[gbXML, 2019] Green Building XML. – <http://www.gbxml.org/index.html>

[GREEN ZOOM, 2019] GREEN ZOOM. – <https://www.greenzoom.ru/>

[Gyrard, 2012] Gyrard A., Bonnet C., Boudaoud K. Enrich Machine-to-Machine Data with Semantic Web Technologies for Cross-Domain Applications // Proc. IEEE World Forum on Internet of Things (WF-IoT), 2014.

[IFC, 2019] IFC-Industry Foundation Classes. – http://www.ifcwiki.org/index.php?title=IFC_Wiki

[IFCtoRDF, 2019] IFCtoRDF-Desktop. – <https://github.com/jyrkioraskari/IFCtoRDF-Desktop>

[Kofler, 2012] Kofler M. J., Reinisch C., Kastner W. A semantic representation of energy-related information in future smart homes // Energy and Buildings, vol. 47, 2012.

[Pauwels et al., 2014] Pauwels P., Corry E., O'Donnell J. Representing SimModel in the Web Ontology Language // Proc. International Conference on Computing in Civil and Building Engineering, Proceedings, 2014.

[Pauwels et al., 2016] Pauwels P., Terkaj W. EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology // Automation in Construction, № 63, 2016.

[Semantic Web, 2019] Semantic Web-W3C. – <https://www.w3.org/standards/semanticweb/>

[SPARQL, 2019] SPARQL Query Language for RDF. – <https://www.w3.org/TR/rdf-sparql-query/>

[SSN, 2019] Semantic Sensor Network Ontology. – <https://www.w3.org/2005/Incubator/ssn/ssnx/ssn>

[Stavropoulos et al., 1975] Stavropoulos T. G., Vrakas D., Vlachava D., Bassiliades N. Bonsai: A smart building ontology for ambient intelligence // Proc. 2Nd International Conference on Web Intelligence, Mining and Semantics, USA. 2012.

[Xue, 2012] Xue L., Liu Y., Zeng P., Yu H., Shi Z. An Ontology based Scheme for Sensor Description in Context Awareness System // Proc. IEEE International Conference on Information and Automation, 2015.

УДК 004.896; 004.75

ПОДДЕРЖКА ФУНКЦИОНИРОВАНИЯ РАСПРЕДЕЛЕННЫХ САПР В «ТУМАННЫХ» СРЕДАХ¹

В.М. Курейчик (*vmkureychik@sfnedu.ru*)
И.Б. Сафроненкова (*safronenkova050788@yandex.ru*)

Федеральное государственное автономное образовательное учреждение высшего образования «Южный федеральный университет», Институт компьютерных технологий и информационной безопасности (ИКТИБ), Таганрог, Россия

Работа посвящена вопросам эффективного функционирования распределенных систем автоматизированного проектирования (РСАПР) в «туманных» средах. В качестве главного критерия эффективности РСАПР выбрано их быстродействие, которое обеспечивается многими факторами, в том числе скоростью решения задачи переноса вычислительной нагрузки. Авторами предложен метод, в основе которого лежит онтологический анализ структур используемых популяционных алгоритмов и способов их разбиения. Принципиальным отличием данного метода является учет специфики информационных обменов между процессами, происходящими в алгоритмах, используемых в РСАПР. Научная новизна данной работы заключается в использовании моделей онтологий для формирования ограничений в задаче переноса вычислительной нагрузки в среде «туманных» вычислений.

Ключевые слова: распределенные системы автоматизированного проектирования, задача переноса вычислительной нагрузки, параллельный алгоритм, онтология, «туманные» вычисления.

Введение

Использование интеллектуальных распределенных систем автоматизированного проектирования (РСАПР) в проектной деятельности обусловлено высокой трудоемкостью задач большой размерности, возникающих в различных областях науки и техники [Каляев и др., 2008]. Примером такой задачи может послужить задача компоновки при

Работа выполнена при финансовой поддержке РФФИ (проект № 18-07-00050, № 18-29-22019)

проектировании СБИС, которая относится к классу NP-полных и NP-сложных задач. Архитектура РСАПР позволяет производить параллельные вычисления, что потенциально сокращает время, необходимое для решения таких задач [Глушань и др., 2015].

К настоящему времени парадигма «облачных» вычислений была расширена до границ сети и получила название «туманных» вычислений. Одной из отличительных особенностей данной концепции является возможность динамического переноса части вычислительной нагрузки из «облачного» слоя в «туманный» и обратно. Это обеспечивает снижение нагрузки на коммуникационную инфраструктуру сети, что, в свою очередь, позволяет производить вычисления с более высокой скоростью [CISCO, 2015], [Moysiadis et al., 2018], [Melnik et al., 2017].

В работах [Мельник и др., 2019a], [Кузюрин и др., 2014] предложены новые подходы к постановке и решению задачи «разгрузки» устройств в рамках «туманной» архитектуры сети, однако не учитывают специфику РСАПР при решении конструкторских задач, главным критерием эффективного функционирования которых является быстродействие.

В связи с этим, становится актуальным вопрос о функционировании РСАПР в «туманных» средах, а именно о разработке новых методов, направленных на сокращение времени решения задачи переноса вычислительной нагрузки.

Авторы настоящей работы предлагают использовать онтологический подход к решению задачи переноса вычислительной нагрузки, который заключается в формировании ограничений на множество узлов-кандидатов. Модель онтологии позволяет учитывать специфику структуры параллельных алгоритмов, применяемых для решения конструкторских задач в РСАПР, и, таким образом, сократить множество потенциальных узлов для размещения нагрузки. Следовательно сокращается время, необходимое для решения задачи переноса вычислительной нагрузки и для решения вычислительной задачи САПР в целом.

1. Постановка задачи переноса вычислительной нагрузки

Задача переноса вычислительной нагрузки сводится к классической задаче построения расписаний [Конвей и др., 1975]. Однако, особенности «туманных» сред, такие как наличие транзитных участков цепи, оказывающих влияние на коммуникационные свойства сети и беспрецедентное количество узлов «туманного» слоя не позволяют использовать данную математическую модель в полной мере.

Математическая модель задачи переноса вычислительной нагрузки, предложенная в [Мельник и др., 2019b] учитывает особенности «туманных» сред. Перейдем к ее формальной постановке.

Пусть дан граф вычислительной задачи G , с некоторой вычислительной сложностью подзадачи x_i и объемом информации w_i , передаваемой между подзадачами. Граф G разделен на два подграфа G' и G'' . Необходимо разместить подграф G' на вычислительные устройства (ВУ) сегмента сети P' туманного слоя, в то время, как вычислительные задачи подграфа G'' продолжает выполняться на сегменте сети P'' (рис.1)

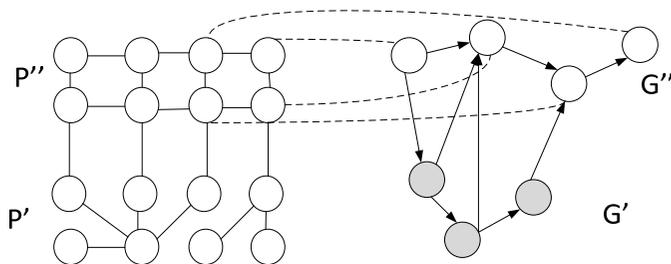


Рис. 1. Распределение вычислительных подзадач для модели «разгрузки» устройств

Полная загрузка j -го вычислительного устройства описывается следующей формулой:

$$L_j = L_{pj}(A) + L_{distj}(A, Flow_in, Flow_out) + L_{nj}(A)$$

где $L_p(A)$ – загрузка узла, порождаемая переносом вычислительной подзадачи на узел;

$L_{dist}(A, Flow_in, Flow_out)$ – загрузка узла, порождаемая обменом информацией между подграфами G' и G'' ;

$L_n(A, Flow_in, Flow_out)$ – загрузка узла, порождаемая передачей информации через узел.

Более подробно с данной формальной постановкой задачи переноса вычислительной нагрузки и методом ее решения можно ознакомиться в работе [Мельник, 2019].

Предложенная модель учитывает особенности «туманных» сред, а именно наличие транзитных участков сети и географическую распределенность вычислительных узлов. Поэтому далее будем использовать данную математическую модель.

2. Исследование структур параллельных популяционных алгоритмов

Использование популяционных алгоритмов для решения NP-сложных комбинаторных задач получило широкое распространение. Во-первых, популяционные алгоритмы обеспечивают высокую вероятность локализации глобального экстремума оптимизируемой функции. Во-вторых, данные алгоритмы позволяют эффективнее классических отыскать субоптимальное решение, которое зачастую оказывается достаточным [Карпенко, 2012].

Ввиду их большого многообразия, используют следующую классификацию в соответствии с признаком их природного прототипа [Карпенко, 2017]:

- эволюционные алгоритмы, включая генетические;
- популяционные алгоритмы, вдохновленные живой природой (алгоритмы роя частиц, колонии муравьев, пчелиного роя, искусственной иммунной системы, бактериальной оптимизации и т.д.);
- алгоритмы, вдохновленные неживой природой (алгоритмы гармонического, гравитационного, электромагнитного поиска и т.д.);
- алгоритмы, инспирированные человеческим обществом (алгоритмы эволюции разума, стохастического диффузионного поиска, «культурные» алгоритмы, меметические алгоритмы и т.д.);
- прочие алгоритмы (самоорганизующийся миграционный алгоритм, алгоритмы рассеянного поиска, прокладки путей и т. д.)

Однако, использование популяционных алгоритмов поисковой оптимизации зачастую требует больших вычислительных ресурсов даже при невысокой вычислительной сложности целевой функции. Следовательно, использование их параллельных прототипов потенциально эффективно для сокращения требуемых вычислительных ресурсов, учитывая природную особенность популяционных алгоритмов к распараллеливанию.

В литературе [El-Ghazali et al., 2008], [Alba, 2005] выделяют следующие модели распараллеливания популяционных алгоритмов:

- глобальная модель;
- островная модель;
- диффузионная модель.

Данные модели распараллеливания широко известны, поэтому не будем на них останавливаться подробно.

Перенос части вычислительной нагрузки подразумевает «отрыв»

сегмента алгоритма от его исходной графовой структуры. Это приводит к процедуре перебора множества вариантов разбиения исходного графа и, как следствие, увеличение времени решения вычислительной задачи в целом. Ввиду этого, необходимо ввести ограничения на способы разбиения исходного графа вычислительных задач, причем для каждой модели параллелизма способ разбиения будет индивидуален.

Популяционные алгоритмы, использующие глобальную модель параллелизма, ориентированы на организацию параллельных вычислений по типу master-slave (мастер-подчиненный).

В данном случае возможны следующие варианты переноса подграфа задач:

- перенос «мастер-процесс» (рис. 2);
- перенос «подчиненный-процесс» (рис. 3);
- перенос «мастер-подчиненный-процесс» (рис. 4).

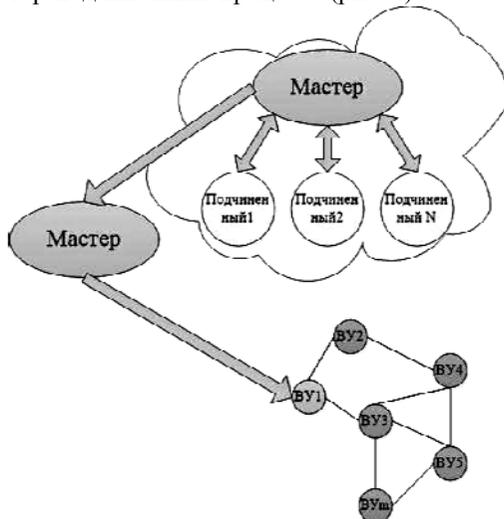


Рис. 2. Схема переноса сегмента «мастер-процесс»

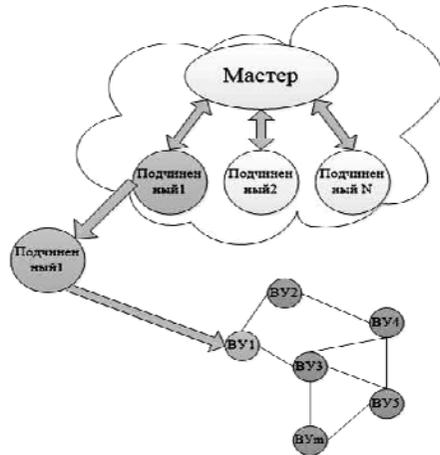


Рис. 3. Схема переноса сегмента «подчиненный-процесс»

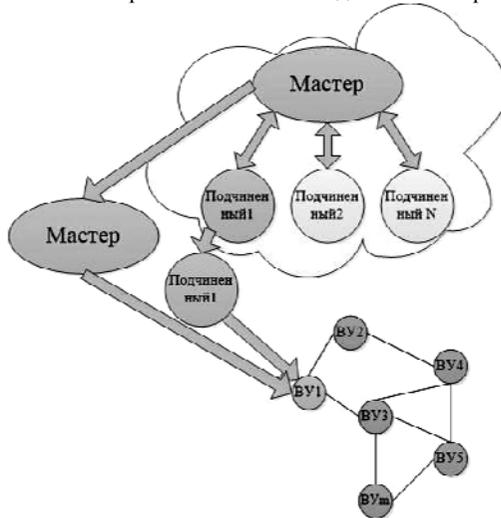


Рис. 4. Схема переноса сегмента «мастер-подчиненный-процесс»

Миграционные алгоритмы, основанные на островной модели параллелизма, в некоторых источниках также называют алгоритмами коммутирующей мультипопуляции. В данном случае разумно сформулировать лишь одно правило разбиения подграфа – перенос острова (популяции) (рис 5).

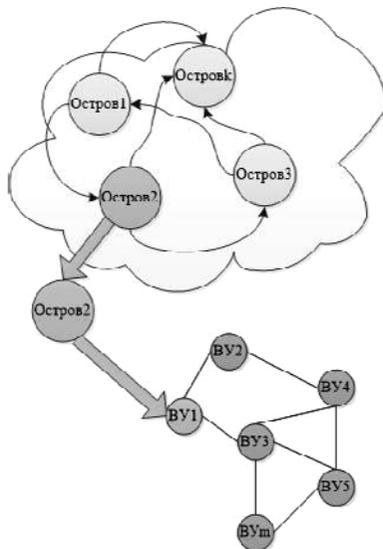


Рис. 5. Схема переноса сегмента «остров»

Алгоритмы, в основе которых лежит диффузная модель параллелизма, можно рассматривать как частный случай островной модели параллелизма. В силу этого факта производить разбиение графовой структуры диффузной модели параллелизма не имеет смысла.

3. Онтологический подход к формированию ограничений в задаче переноса вычислительной нагрузки

Метод решения задачи переноса вычислительной нагрузки, предложенный в работе [Kalyaev et al., 2019] является универсальным для любых алгоритмов, используемых в распределенных системах. Однако, он носит итерационный характер, и в наихудшем случае его исполнения, учитывая огромное количество узлов «туманного» слоя, невозможно предсказать время его работы. Такой исход является неудовлетворительным для случая функционирования РСАПР, ввиду соблюдения критерия быстродействия.

В связи с этим, авторы предлагают метод, позволяющий учитывать специфику структур параллельных алгоритмов, тем самым, сократить количество потенциальных узлов для размещения вычислительной нагрузки. В основе данного метода лежит модель онтологии, позволяющая формально описать знания о структурах параллельных популяционных алгоритмах и способах их разбиения. Наряду с информацией о ресурсах потенциальных для размещения узлов

происходит усечение множества узлов-кандидатов для переноса вычислительной нагрузки. Концепция предложенного метода изображена на рис. 6.



Рис. 6. Концепция метода формирования ограничений в задаче переноса вычислительной нагрузки

Идея формирования ограничений на множество узлов-кандидатов для переноса вычислительной нагрузки заключается в проведении процедуры онтологического анализа и использовании сведений об узлах-кандидатах для переноса вычислительной нагрузки. К сведениям о ресурсах узлов-кандидатов относятся загруженность узла, его производительность, удаленность от узла-лидера, находящегося в «облаке» и отвечающего за перенос вычислительной нагрузки.

Согласно предложенному методу, исходные данные популяционного параллельного алгоритма, позволяющие учесть его специфику информационных обменов, происходящих на уровне процессов, могут быть описаны в модели онтологии.

Для реализации предложенного онтологического подхода к формированию ограничений в задаче переноса вычислительной нагрузки необходимо принять следующие допущения:

1. каждый узел «туманного» слоя обладает информацией о структуре используемого алгоритма;
2. множество узлов «туманного» слоя достаточно большое, но ограниченное;
3. существует конечное множество способов разбиения параллельных алгоритмов, используемых для решения NP-сложной задачи в РС и оно может быть описано в виде онтологии.

4. Формализация знаний о структуре параллельных популяционных алгоритмов

Неотъемлемой частью любой интеллектуальной автоматизированной системы является формализованное знание о предметной области. Согласно определению, формализация знаний – это представление знаний субъекта о предметной области с использованием знаковых систем [Гаврилова и др., 2005]. В настоящее время существует несколько способов представления знаний в области искусственного интеллекта: продукционная модель, семантические сети, фреймы, логика дескрипций [Сидоркина, 2015] и онтологии [Цуканова, 2014]. Многие авторы считают использование онтологий для представления знаний в РС перспективной технологией [Добров и др., 2014], [Kureychik, 2015], [Загорулько и др., 2016], [Kureichik, et al., 2017]. Онтологии позволяют интегрировать и использовать неструктурированную информацию и знания, формализовать различные предметные области, осуществлять сложный логический вывод, позволяющий автоматически получать новые знания и контролировать их согласованность. Тем самым онтологии представляют собой надёжные средства с точки зрения обеспечения согласованности, повторного использования и совместимости моделей представления знаний в интеллектуальной РС.

В рамках настоящей работы предлагается использовать модель онтологии для формализации знаний о структуре параллельных моделей популяционных алгоритмов, используемых для решения NP-сложных задач в РСАПР, и способов их разбиения. Это позволяет учесть специфику применяемых алгоритмов с точки зрения частоты и объемов информационных обменов между процессами, и, таким образом, сократить множество узлов, заведомо не подходящих для размещения вычислительной нагрузки. Иначе говоря, используя модель онтологии, возможно сократить множество узлов-кандидатов для проведения процедуры моделирования размещения, и, соответственно, сократить общее время решения задачи размещения вычислительной нагрузки.

Анализ используемых популяционных алгоритмов, их структур и способов разбиения позволяет составить формализованное описание рассматриваемой предметной области (рис. 7).

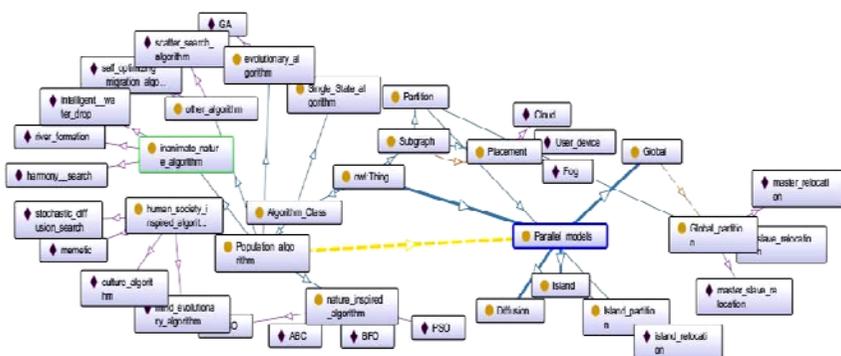


Рис. 7 Модель онтологии структур параллельных популяционных алгоритмов и способов их разбиения.

Таким образом, разработанная модель онтологии содержит следующие формализованные знания:

- класс популяционного алгоритма;
- модель распараллеливания в соответствии с классом;
- ограниченное множество способов разбиения алгоритмической структуры.

На основе данных сведений происходит усечение множества узлов-кандидатов для размещения вычислительной нагрузки при частичном ее переносе из «облака» в «туманный» слой. Таким образом, сокращается общее время решения задачи переноса вычислительной нагрузки в РСАПР, что повышает ее быстродействие.

Заключение

В работе предложен метод поддержки функционирования распределенных САПР в среде «туманных» вычислений. Рассмотрены формальные постановки задачи переноса вычислительной нагрузки. Обозначены их недостатки в рамках функционирования РСАПР в «туманной» среде с учетом критерия быстродействия. Проведенный анализ популяционных алгоритмов и моделей их распараллеливания выявил наличие характерных особенностей, заключающихся в объемах и частоте информационных обменов между процессами, присущие каждой модели. Разработана модель онтологии, содержащая формализованные знания о структуре рассмотренных алгоритмов. На основании этих знаний, узлы, потенциально подходящие для размещения с точки зрения

имеющихся ресурсов, но, не отвечающие требованиям специфики информационных обменов между процессами, не рассматриваются в качестве кандидатов для размещения вычислительной нагрузки. Таким образом, множество узлов для проведения процедуры моделирования размещения сокращается. Соответственно, сокращается общее время решения задачи переноса вычислительной нагрузки в РСАРР.

Список литературы

[Гаврилова и др., 2005] Гаврилова Т. А., Кудрявцев Д. В., Муромцев Д. И. Инженерия знаний. Модели и методы: Учебник. — СПб.: Издательство «Лань», 2016.

[Глушань и др., 2015] Глушань В. М., Лаврик П.В. Распределенные САПР. Архитектура и возможности. – Старый Оскол: ТНТ, 2015.

[Добров и др., 2014] Добров Б.В., Иванов В.В., Лукашевич Н.В., Соловьев В.Д. Онтологии и тезаурусы: модели, инструменты, приложения: учеб. пособие. – М.: ИнтернетУниверситет Информационных Технологий; БИНОМ. Лаборатория знаний, 2013.

[Загорюлько и др., 2016] Загорюлько Ю.А., Загорюлько Г.Б., Боровикова О.И. Технология создания тематических интеллектуальных научных интернет-ресурсов, базирующаяся на онтологии // Программная инженерия. – М.: Новые технологии, 2016, № 2.

[Каляев и др., 2008] Каляев И.А., Левин И.И., Семерников Е.А., Шмойлов В.И. Реконфигурируемые мультиконвейерные вычислительные структуры. – Ростов-на-Дону: ЮНЦ РАН, 2008.

[Карпенко, 2012] Карпенко А. П. Популяционные алгоритмы глобальной поисковой оптимизации. Обзор новых и малоизвестных алгоритмов // Информационные технологии. – М.: Новые технологии, 2012. № 7.

[Карпенко, 2017] Карпенко А.П. Основные сущности популяционных алгоритмов глобальной оптимизации. Опыт систематизации // Интернет-журнал «НАУКОВЕДЕНИЕ», 2017, 9(7).

[Конвей и др., 1975] Конвей Р.В, Максвелл В.Л., Миллер Л.В. Теория расписаний. М.: Наука, 1975, 360 с.

[Кузюрин и др., 2014] Кузюрин Н.Н, Грушин Д.А., Фомин С.А. Проблемы двумерной упаковки и задачи оптимизации в распределенных вычислительных системах // Труды Института системного программирования РАН, 2014.

[Мельник и др., 2019а] Мельник Э.В., Клименко А.Б., Иванов Д.Я. Модель задачи распределения вычислительной нагрузки для информационно-управляющих систем на базе концепции туманных вычислений // Известия ТулГУ. Технические науки. – М.: ИПУ РАН, 2019.

[Мельник и др., 2019б] Мельник Э.В., Клименко А.Б., Иванов Д.Я. Модель задачи формирования сообществ устройств информационно-управляющих систем в средах туманных вычислений // XIII Всероссийское совещание по проблемам управления ВСПУ-2019 – Тула: Тульский государственный университет, 2018. Вып. 2

[Сидоркина, 2015] Сидоркина И.Г. Системы искусственного интеллекта: учебное пособие / И.Г. Сидоркина. – М.: КНОРУС, 2015.

[Цуканова, 2014] Цуканова Н.И. Онтологическая модель представления и организации знаний. Учеб. пособие для вузов. – М.: Горячая линия – Телеком, 2014.

[Alba, 2005] Alba E. Parallel metaheuristics. A New Class of Algorithms. – NY: John Wiley, 2005.

[CISCO, 2015] Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are – https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf.

[El-Ghazali et al., 2008] El-Ghazali T. et al. Parallel Approaches for Multiobjective Optimization // Multiobjective Optimization. – Berlin Heidelberg: Springer-Verlag 2008.

[Kalyaev et al., 2019] Kalyaev I., Melnik E., Klimenko A. A Technique of Adaptation of the Workload Distribution Problem Model for the Fog-Computing Environment. In: Silhavy R. (eds) Cybernetics and Automation Control Theory Methods in Intelligent Algorithms. CSOC 2019. Advances in Intelligent Systems and Computing, vol 986. Springer, Cham, 2019.

[Kureychik, 2015] Kureychik V.M. Overview and problem state of ontology models development // Conference Proceedings IEEE 2015 9th International Conference on Application of Information and Communication Technologies (AICT), 14-16 Oct 2015, Rostov-on-Don, Russia, 2015

[Kureichik et al., 2017] Kureichik V., Safronenkova, I. Ontology-Based Decision Support System for the Choice of Problem-Solving Procedure of Commutation Circuit Partitioning. In: Kravets, A., Shcherbakov, M., Kultsova, M., Groumpos, P. (Eds.) Creativity in Intelligent Technologies and Data Science. CCIS, Cham: Springer, 2017, vol. 754.

[Melnik et al., 2017] Melnik, E. V. and Klimenko, A. B. (2017) ‘Informational and control system configuration generation problem with load-balancing optimization’, in Application of Information and Communication Technologies, AICT 2016 - Conference Proceedings.

[Moysiadis et al., 2018] Moysiadis V., Sarigiannidis P. Moscholios I. Towards Distributed Data Management in Fog Computing // Wireless Communications and Mobile Computing, 2018.

[Noy et al., 2001] Noy, N., McGuinness, D.: Ontology development 101: a guide to creating your first ontology. Stanford Knowledge Systems Laboratory Technical report KSL-01-05 and Stanford Medical Informatics Technical report SMI-2001- 0880 (2001).

[Protégé, 2019] Protégé –a FreeOntology Editor and Knowledge-Base Framework. –<http://protege.stanford.edu>.

ИНФРАСТРУКТУРА IACRAAS ДЛЯ ФОРМИРОВАНИЯ ИНТЕРПРЕТИРУЕМЫХ БАЗ ДИАГНОСТИЧЕСКИХ ЗНАНИЙ ПО ЗАБОЛЕВАНИЯМ ПРОИЗВОЛЬНОЙ НАПРАВЛЕННОСТИ¹

В.В. Грибова (*gribova@iacp.dvo.ru*),
Д.Б. Окунь (*okdm@iacp.dvo.ru*),
М.В. Петряева (*margaret@iacp.dvo.ru*),
Е.А. Шалфеева (*shalf@iacp.dvo.ru*)

Институт автоматики и процессов управления ДВО РАН,
Владивосток

В работе представлена облачная инфраструктура для формирования интерпретируемых баз диагностических знаний для систем поддержки принятия решений в медицине. Лежащая в основе построения всех компонентов таких систем онтология позволяет формализовать любое заболевание как многогранный и многовариантный развивающийся внутренний процесс. Работа может быть полезна разработчикам систем медицинской диагностики.

Ключевые слова: диагностика, заболевание, база терминов и наблюдений, база знаний, онтология, редактирование знаний, основанная на знаниях система.

Введение

Продолжающаяся информатизация медицины уже позволила добиться успехов в организации документооборота и отчасти сделала более цивилизованными отношения пациентов с учреждениями и интересующими их специалистами. Однако эпоха автоматизации поддержки интеллектуальной работы врача еще не наступила. Одна из проблем связана с особенностью знаний, используемых врачами: они основаны на наиболее сложно устроенных причинно-следственных связях, их объем чрезвычайно велик, в некоторых сферах растет или

¹ Работа выполнена при частичной финансовой поддержке РФФИ (проекты №18-07-01079 и №17-07-00956) и программы «Приоритетные научные исследования в интересах комплексного развития ДВО РАН» (проект №18-5-078).

обновляется лавинообразно [Рыбина, 2018, Ле, 2014, Livemd.ru, 2017].

Интеллектуальные системы для диагностики и прогноза, основанные на нейросетях, уже достигают обнадеживающего процента точности. Но доверять их результатам на практике недопустимо, т.к. решение должно быть понятно врачу, а такие системы не предоставляют объяснение.

Интеллектуальные системы, основанные на прецедентах, выдают не слишком убедительный результат, поскольку набор показателей пациента объемом в несколько десятков почти всегда уникален и не обязательно заполнен целиком. Кроме того, работа с временными рядами, отражающими процесс развития заболевания у пациента, практически не поддерживается. Тем не менее для задач диагностики и прогноза иногда удается получать результаты, которые полезны врачу, принимающему или анализирующему решение.

Для построения систем, основанных на знаниях, способных выдавать аргументированный совет или решение, принципиально важны адекватные современному уровню базы знаний (БЗ). Но врачам или экспертам все еще не предложены средства формирования знаний в привычных терминах и в естественной структуре, так чтобы прописать для заболевания симптомы, синдромы, множество факторов, нозологические формы, этиологии, патогенез, клинические проявления для разных категорий пациентов, и чтобы именно эти знания использовались средствами поддержки решений.

За несколько десятилетий установлено и апробировано, что модель представления знаний о заболеваниях (онтология) не зависит (инвариантна) от медицинского профиля. К настоящему времени предложена парадигма и реализована облачная технология редактирования информации на основе явно описываемой структуры этой информации, используемой терминологии предметной области (прОбл) и ограничений интерпретации, а также технология создания программных сервисов поддержки и автоматизации работы специалистов по обработке этой информации.

Целью работы является представление облачной инфраструктуры для формирования интерпретируемых баз диагностических знаний - компонентов систем поддержки принятия решений (СППР) в медицине.

1. Онтология медицинских знаний

Одной из онтологий медицинской диагностики, соответствующей реальным представлениям медицины и широкому спектру заболеваний, является «Онтология медицинской диагностики заболеваний» [Gribova, 2018]. Она содержит набор отношений между понятиями для описания

клинической картины заболеваний в динамике патологического процесса, а также с учетом воздействия лечебных мероприятий и других событий на проявления заболеваний.

В онтологии медицинской диагностики определены типы утверждений (позволяющих решать задачи диагностики), такие как <заболевание-k, признак-j, диапазон значений-kj признака-j>; «необходимое условие существования заболевания», вариант симптомокомплекса для заболевания; «вариант-m процесса изменения значений-kj признака-j для некоторого симптомокомплекса», условия, необходимые для возникновения заболевания; вариант реакции на воздействие события и т.д. Эти типы утверждений позволяют сформировать базу знаний, достаточную и для поиска гипотез о возможных единичных диагнозах, и для выявления сочетанной и осложненной патологий, и для сокращения множества гипотез о диагнозе. (Последняя решается через поиск запроса такой дополнительной информации о состоянии пациента, ответ на который опровергнет один из ранее предполагаемых диагнозов).

Онтология медицинской диагностики обладает следующими характеристиками.

1. Возможность формирования симптомокомплексов заболеваний с учетом категорий пользователей с использованием референсных диапазонов вместо определенных "норм" для лабораторных и инструментальных показателей.

2. Возможность формирования альтернативных симптомокомплексов с разными подходами к выявлению достоверных признаков заболевания, чтобы в процессе диагностики выбрать наиболее щадящий быстрый или недорогой.

3. Возможность уточнения диагнозов с учетом этиологии, патогенеза, варианта течения и пр. для проведения дифференциальной диагностики заболеваний и выбора соответствующих методов лечения.

4. Единообразная формализация стадий хронических болезней и периодов развития острых заболеваний.

5. Возможность указания меры нечёткости признака в некотором периоде динамики: модальность, значения которой (необходимо, характерно, возможно) достаточны для рекомендаций по подтверждению или опровержению гипотезы; шкала вероятностей соответствия признаков гипотезе о диагнозе, которая может быть использована, если эксперты прОбл имеют соглашения о том, какие ее значения целесообразно использовать при принятии решений).

6. Учет значений характеристик и признаков, измененных воздействием событий. Наличие такого элемента причинно-следственных

связей позволяет учитывать внешние воздействия, оказываемые на организм пациента на разных этапах заболевания.

7. Учет разных вариантов динамики значений признаков. Такие элементы знаний позволяют учитывать (на основании врачебного опыта) многообразие течения одних и тех же заболеваний у разных пациентов.

Признак заболевания может быть простым или составным, его значения представляются по *периодам динамики* развития признака или заболевания в целом; могут задаваться необходимые условия для рассмотрения признака. Каждый *период динамики* характеризуется *верхней* и *нижней границей длительности периода, единицей измерения границ*. Составной признак содержит описание наборов своих изменяющихся по периодам характеристик с модальностью вхождения характеристики признака в клиническую картину.

2. Основные требования к инструментам формирования интерпретируемых баз знаний

Создание интерпретируемых баз знаний возможно при формировании их на основе онтологии, которая задает правила порождения информации, и ограничения ее интерпретации.

Онтология становится структурной основой средств редактирования. Соответственно, *среда для создания* интерпретируемых баз данных должна включать **Редактор онтологии**, который позволит инженерам знаний (когнитологам) вносить, в случае необходимости, изменения в онтологию.

Такая среда должна предоставлять **Редактор баз знаний**, управляемый онтологией и ориентированный на экспертов предметной области. Учитывая разный уровень пользователей-экспертов, различную структуру формируемых знаний, редактор БЗ должен иметь развитые средства интерфейсной поддержки пользователя: возможность просматривать с разной степенью детализации и в разной форме фрагменты базы знаний, контролировать правильность вводимой информации с учетом ограничений, определенных в онтологии, полноту формирования базы знаний.

Базы знаний должны формироваться в единых терминах предметной области для обеспечения их однозначной интерпретации как программными сервисами, так и различными экспертными сообществами. Для обеспечения этого требования инфраструктура для формирования баз знаний должна включать *Терминологический справочник*, формирование которого, также как и базы знаний, должно осуществляться на основе онтологии с использованием редактора терминов. В этом случае при

формировании утверждений в базе знаний термины (признаки и факторы, их значения) должны не определяться в базе знаний, а выбираться из терминологического справочника. Наличие онтологии, ориентированной на экспертов, использование единого терминологического справочника позволяет обеспечить важное свойство баз знаний – их интерпретируемость.

Ввиду важности качества баз знаний требуется *Средства оценивания* их качества на эталонных архивах решенных задач.

Совокупность *Редактора онтологии, Редактора БЗ, Средства оценивания* качества Баз знаний на эталонных архивах решенных задач, *Редактор терминологической* базы образуют *инфраструктуру формирования интерпретируемых баз диагностических знаний*.

Ввиду важности эволюционированности баз знаний требуется инструментарий для непрерывного развития (совершенствования) баз знаний не только экспертами предметной области, но и автоматически (с последующей верификацией экспертами) на основе новых данных (историй болезни) - средства индуктивного формирования Баз знаний или фрагментов БЗ, интегрируемых с ней, средства адаптации Баз знаний к изменениям знаний в предметной области (когда практические задачи приносят эти знания). При этом каждый акт адаптации потребует использования средства проверки качества новой версии БЗ (для подтверждения, что ранее решенные задачи с ее помощью решаются не хуже).

Добавление к отмеченным выше инструментам инфраструктуры средства индуктивного формирования Баз знаний и средства адаптации БЗ превращают инфраструктуру формирования интерпретируемых баз диагностических знаний в *специализированную среду их развития*.

3. Характеристика инструментов IACPaas для формирования знаний и их обновления

На платформе IACPaas реализован комплекс инструментов для обработки информационных ресурсов, формируемых по явно представляемой онтологии. Форма представления обоих уровней - иерархические семантические сети. Иерархические семантические сети IACPaas достаточны для описания вышперечисленных связей (между названиями признаков, диапазонами их значений и т.д.).

На платформе IACPaas имеются возможности для расширения комплекса инструментов для обработки информационных ресурсов в полномасштабную *специализированную среду* - инфраструктуру для формирования и *развития* интерпретируемых баз диагностических знаний.

Штатный Редактор онтологий обеспечивает простое и удобное создание онтологий на метаязыке [Грибова 2015]. Генератор редакторов знаний платформы IACPaaS автоматически генерирует редактор БЗ, при этом поддерживается три типа пользовательского интерфейса для редактирования и просмотра базы знаний (пользователь может выбрать любой). Сгенерированный редактор знаний обеспечивает процесс создания баз знаний на основе онтологии, контролируя ограничения на формирование знаний и обеспечивая контроль ее целостности и полноты. Остальные инструменты добавляются на платформу по мере готовности.

Для реализации процесса монотонного усовершенствования баз знаний (на основе метода индуктивного формирования фрагментов БЗ) требуется средство поддержки выбора прецедентов (правильно решенных задач).

Т.о. комплекс инструментов IACPaaS для формирования БЗ и их развития таков: IACPaaS-Редактор онтологии, Генератор редакторов баз знаний с выбором типа пользовательского интерфейса, IACPaaS-Редактор БЗ, генерирующийся по онтологии с само-адаптирующимся интерфейсом (при изменении онтологии), Редактор терминологической базы, Средство проверки качества знаний на контрольных примерах (известных правильно решенных задачах), которое, по существу, является специализированным решателем задачи диагностики, Средство индуктивного формирования новой версии БЗ, Средство для выбора прецедентов для Усовершенствования БЗ).

4. Формирование знаний с поддержкой их обновления инструментами IACPaaS

С помощью инструментария, позволяющего создавать интерпретируемые БЗ по онтологии, к настоящему времени сформированы базы знаний по диагностике заболеваний из различных разделов медицины: группа заболеваний желчного пузыря, желчевыводящих путей и поджелудочной железы; нарушения обмена глюкозаминогликанов; геморрагические лихорадки; заболеваний органов дыхания, органов зрения. [Кобринский 2018; Компанец 2018, Москаленко 2012; Москаленко 2011].

Заболевания представлены несколькими *симптомокомплексами*, соответствующими разному течению заболевания в разных возрастных группах пациентов или разным методикам диагностирования. В комплексах *жалоб, объективных исследований, лабораторных и инструментальных исследований* представлены множества *признаков, изменения значений* которых являются симптомами заболевания.

Детализация диагноза представляет дополнительный набор признаков, позволяющий внести соответствующее уточнение к основному диагнозу с учетом этиологии, патогенеза, варианта течения, стадии и т.д.

После формализации знаний о каждом заболевании на основе экспертных знаний (формировании их в базе знаний) проводится проверка ее соответствия эталонным решениям из архива медицинского учреждения. Используется программный сервис, который анализирует каждый симптом в эталоне на соответствие БЗ и выдает подробный отчет.

Если требуется обновление базы знаний в связи с получением множества «фактов» (прецедентов, решений), не согласованных со знаниями (когда прецедент из врачебной практики содержит правильный результат решения задачи, который не соответствует результату, вырабатываемому системой поддержки решений на основе текущей БЗ, то эффективно будет автоматически (индуктивными методами) формировать новую версию БЗ (и оценить согласованность СППР по обновленной БЗ с имеющимися прецедентами).

Согласно принципу монотонного усовершенствования баз знаний адаптация требуется чаще, чем целесообразно использовать индукцию. При получении новых знаний (утверждений), а также при получении отдельных фактов из практики, свидетельствующих об обновлении вариантов клинической картины известных заболеваний, естественно модифицировать БЗ вручную ради согласования с имеющимися фактами. Для этого процесса требуются Редактор БЗ и программный сервис проверки соответствия новой версии эталонным решениям (или Средство проверки качества новой версии БЗ: проверки не-ухудшения правильности решения множества эталонных задач при замене версии БЗ). Способ проверки неухудшения правильности знаний: подача на вход решателю, интегрированному с новой версией БЗ, входных условий эталонной задачи и сравнение полученного объяснения с выходным результатом эталонной задачи. Пример такого обновления знаний «от практики»: из (в) медучреждения(и) через определенный промежуток времени «приходит» правильный результат диагностики или лечения, он сравнивается с результатом (объяснением гипотез) от СППР: не противоречат ли они друг другу.

Наличие и работоспособность инструментария для реализации процесса монотонного усовершенствования баз знаний (на основе методов адаптации и индуктивного формирования) обеспечивает свойство «Обновляемость знаний».

Среда развития знаний предоставляет Редактор онтологии. Внесение дополнительных понятий или новых связей между понятиями в

онтологию не нарушает целостности Базы знаний, при этом ПИФ эксперта адаптируется на платформе IASaaS автоматически.

Заключение

Предложенная инфраструктура формирования интерпретируемых баз медицинских диагностических знаний включает следующие средства: редактор БЗ, генерирующийся по онтологии с само-адаптирующимся пользовательским интерфейсом (при изменении онтологии), редактор терминологической базы, средства проверки качества знаний, индуктивного формирования новой версии БЗ (в настоящее время находится в стадии разработки), выбора прецедентов для усовершенствования БЗ.

Лежащая в основе построения знаний онтология позволяет формализовать любое заболевание как многогранный и многовариантный развивающийся внутренний процесс.

Инфраструктура реализуется на платформе IASaaS и уже используется специалистами для создания баз знаний в различных областях медицины. Эксперты, заинтересованные в накоплении, улучшении и применении знаний о диагностике имеют возможность присоединиться к этому процессу (как для собственных баз, так и для коллективно используемых ресурсов).

Список литературы

[Рыбина, 2018] Рыбина Г.В., Демидов Д.В., Фонталина Е.С. Автоматизированное получение, представление и обработка темпоральных знаний в динамических интегрированных экспертных системах // Шестнадцатая национальная конференция по искус. инт. КИИ-2018. Труды конференции. Т.1.-Вороново, Москва: НИУ «ВШЭ», 2018. С. 61-69.

[Ле, 2014] Ле Н.В., Камаев В.А., Панченко Д.П., Трушкина О.А. Модель представления знаний при создании медицинской экспертной системы дифференциальной диагностики // Известия Волгоградского государственного технического университета. - 2014. - № 6. - С. 42-50.

[Livemd.ru, 2017] Искусственный интеллект для Медицины – обзор. – [Электронный ресурс]. - http://www.livemd.ru/tags/iskusstvennyj_intellekt.

[Gribova, 2018] Gribova V., Okun D., Petryaeva M., Shalfeeva E., Tarasov A. (2018) Ontology for Differential Diagnosis of Acute and Chronic Diseases. In: Kuznetsov S., Osipov G., Stefanuk V. (eds) Artificial Intelligence. RCAI 2018. Communications in Computer and Information Science, vol 934. Springer, Cham – Pp. 152-163.

[Грибова 2015] Грибова В.В., Клещев А.С., Москаленко Ф.М., Тимченко В.А. Двухуровневая модель сложноструктурированных информационных единиц,

соответствующая метафоре анкетирования // Научно-техническая информация. Сер. 2. 2015. № 10. С. 1-10.

[Kobrinskii 2018] Kobrinskii B.A., Demikova N.S., Blagosklonov N.A. (2018) Knowledge Engineering in Construction of Expert Systems on Hereditary Diseases. In: Kuznetsov S., Osipov G., Stefanuk V. (eds) Artificial Intelligence. RCAI 2018. Communications in Computer and Information Science, vol 934. Springer, Cham.

[Компанец, 2018] Компанец Г.Г., Иунихина О.В., Петряева М.В., Шалфеева Е.А., Окунь Д.Б. Новые подходы к диагностике геморрагической лихорадки с почечным синдромом: разработка технологии дистанционной диагностики на облачной платформе // Глобальный научный потенциал, 2018. №12 (93). С. 80-83.

[Москаленко, 2012] Москаленко Ф.М., Черняховская М.Ю. Формальное описание заболевания Хронический панкреатит // Информатика и системы управления. 2012. №4. С. 97-106.

[Москаленко, 2011] Москаленко Ф.М., Черняховская М.Ю. База заболеваний "конъюнктивит" для компьютерного банка знаний // Информатика и системы управления. 2011. № 2 (28). С. 50-58.

УДК 004.896

ВЫБОР АРХИТЕКТУРЫ ГЛУБОКИХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ОТСЛЕЖИВАНИЯ ДИНАМИКИ СОСТОЯНИЯ ОБЪЕКТА¹

М.И. Дли (*midli@mail.ru*)

А.Ю. Пучков (*putchkov63@mail.ru*)

Е.И. Лобанева (*lobaneva94@mail.ru*)

М.А. Василькова (*vasilkova_mariya00@mail.ru*)

Филиал ФГБОУ ВО «Национальный исследовательский университет «МЭИ» в г. Смоленске

В работе предложена архитектура глубокой нейронной сети для отслеживания динамики состояния сложного технологического объекта по данным, поступающих в форме изображений, а также рекомендации по ее адаптации под конкретное применение. В основе разработанной архитектуры лежит каскадное использование сверточных нейронных сетей для обработки многоканальной видеoinформации из разных технологических зон одного объекта.

Ключевые слова: машинное обучение, сверточные нейронные сети, компьютерное зрение

Введение

Глубокие нейронные сети (DNN, deep neural networks) сейчас представляют наиболее практически значимое направление развития методов искусственного интеллекта. Особенно активно DNN применяются в системах видеоаналитики для получения метаданных из видеопотока. С технической точки зрения видеоаналитика – это программно-аппаратный комплекс интеллектуального анализа событий, попадающих в сектор захвата систем видеонаблюдения и

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 19-01-00425.

подлежащих глубокой обработке программными средствами. В среднем используется только 10% данных, которые способна предоставить к обработке видеочкама, а интеллектуализация этого процесса позволит увеличить долю полезного использования информации.

Первоначально видеоаналитика стала применяться в системах производственной безопасности для обнаружения движения объекта или пересечения контрольной линии, идентификация объектов (люди, транспорт, багаж), оценка их поведения [Рубио, 2016]. Однако этой областью применения не ограничиваются возможности использования компьютерного зрения на производстве. В рамках утвержденной Правительством Российской Федерации в 2017 году программы развития «Цифровая экономика Российской Федерации» можно отметить следующие перспективные направления автоматизации обработки видеoinформации в автоматизированных системах управления технологическими процессами (АСУТП) с использованием методов искусственного интеллекта:

- реидентификация объектов и процессов при переходе их из одной производственной зоны в другую в соответствии с принятыми технологическими и логистическими цепочками;

- автоматизированный контроль выполнения требований техники безопасности;

- решение транспортных задач в условиях большой размерности исходных данных, когда методы линейного программирования приводят к значительным временным затратам;

- поиск и прогноз неисправностей оборудования для снижения вероятности возникновения аварийных ситуаций;

- реализация многомерного машинного зрения на основе обработки информации с большого числа датчиков с целью наблюдения за технологическими процессами (ТП) в реальном времени и прогнозирования их поведения;

- бизнес-аналитика на основе генерации интеллектуальными видеочкамами метаданных о состоянии производственного процесса, а также извлечения скрытых закономерностей в визуализированной информации о результатах коммерческой деятельности предприятия.

Большинство отмеченных направлений применения методов искусственного интеллекта предполагает обработку видеоданных, в том числе оценку скорости изменения состояния объекта наблюдения. Изменение изображения исследуемого объекта можно применять как визуальную подсказку в процессе принятия решений, тем более что смысл

многих действий именно в динамике – бывает достаточно пронаблюдать движение отдельных точек, чтобы распознать событие. При этом под изменениями может пониматься широкий набор характеристик изображения – смещение каких-либо объектов и контуров на фоне, изменение яркости элементов, модификация текстуры [Li et al., 2016].

Значительное разнообразие природы наблюдаемых объектов и процессов делает актуальной задачу разработки и адаптации методов и алгоритмов машинного зрения для оценки скорости изменения состояния ТП с учетом особенностей производства.

1. Постановка задачи

Сложные ТП характеризуются значительной протяженностью не только во времени, но и в пространстве, что обуславливает целесообразность включения в информационные контуры АСУТП не только сигналов от штатной контрольно-измерительной аппаратуры, но и дополнительно установленных систем визуального контроля высокоответственных технологических зон [Похабов, 2019].

Пусть имеется kz технологических зон, для наблюдения за которыми установлены видеокамеры. С каждой видеокамеры поступает поток кадров с разрешением $n[ip] \times m[ip]$ пикселей, где $ip=1, 2, \dots, kz$ и частотой f . В результате видеоданные от всех видеокамер отображаются тензором X шестого ранга с формой: камера, образцы, кадры, высота, ширина, цвет [Шолле, 2018].

Разные модели камер позволяют формировать видео с частотой кадров, лежащей в большом диапазоне – обычно от 10 до 60 fps (frames per second). Для дальнейшей обработки нейронной сетью интервал дискретизации по времени $\Delta t(i)$, $i=1, 2, \dots, I$, где I – количество информационных видеоканалов, для i -го информационного канала, выбирается большим, чем одна секунда. Этот выбор базируется на предположении, что инерционность ТП позволяет это сделать без нарушения требований теоремы Котельникова, которая устанавливает максимальную величину интервала дискретизации, при которой возможно точное восстановление исходного непрерывного сигнала. Если интервал дискретизации нужен менее чем одна секунда, то следует более тщательно подходить к выбору модели видеокамеры, но методология излагаемого в дальнейшем подхода к распознаванию динамики не изменится.

Требуется разработать архитектуру глубокой нейронной сети для извлечения из формируемого тензора X метаданных, обеспечивающих распознавание и прогнозирование «движения» (эволюции) ТП во

времени. Оценка качества распознавания будет осуществляться на основе матрицы неточностей *CM* (confusion matrix) [Шунина, 2015].

2. Общая архитектура проекта (framework)

DNN находят применение в алгоритмической поддержке систем видеоаналитики для различных прикладных областей, например: обнаружение технологических дефектов [Cha, 2017], медицинской диагностике [Kogi et al., 2018], обработке сенсорной информации, [Kasnesis et al., 2018], идентификации транспортных средств [Xiang, 2016] и других. Однако в большинстве случаев указанные алгоритмы не поддерживают одновременное описание объекта изображением и движением, что позволило бы распознавать события даже в низком разрешении и прогнозировать эволюцию состояния объекта наблюдения. Исключением здесь можно назвать лишь направление, связанное с распознаванием действий человека [Ahlawat et al., 2019], [Fan et al., 2018], сложность которого связана с необходимостью учета окружения, от которого зависит интерпретация смысла распознаваемого действия.

Один из популярных методов, называемый «скользящее окно» состоит из нескольких этапов: выделение фрагмента (пространственно-временного параллелепипеда); решение задачи классификации изображений и поиска объектов для трехмерного пространственно-временного объема. Однако он не подходит для автоматического распознавания изменений, так как требует предварительного указания границ окна на изображении, что делает затруднительным его применение при необходимости смещения границ этого окна в процессе наблюдения за ТП.

В других методах анализа видеопоследовательностей базовым инструментом является понятие оптического потока. Впервые этот подход был предложен Лукасом и Канаде в 1981 году. Оптический поток часто определяется как векторное поле или изображение видимого движения объектов, поверхностей или краев сцены, получаемое в результате перемещения наблюдателя относительно сцены [Сакович и др., 2014]. В процессе анализа оптического потока для каждого пикселя одного кадра вычисляется вектор смещения от текущего кадра к следующему. В результате производится сопоставление: для каждого пикселя одного кадра находят ту же точку на другом кадре. К недостаткам данного подхода можно отнести необходимость анализа разрешимости задачи оптического потока, что влечет за собой необходимость контроля апертурной проблемы [Нагиев и др., 2017]. На основе отмеченных методов созданы специализированные программные продукты, в

частности, для отображения и измерения скорости потока частиц какой либо среды [Thielicke, 2019].

Предлагаемый алгоритм основан на применении ансамбля *DNN*, в структуру которого включен блок временной задержки для обеспечения возможности распознавания динамики объектов по видеоданным.

3. Предлагаемые решения

Успехи применения глубоких нейронных сетей в системах компьютерного зрения различного назначения дают основания для оптимизма и в случае распознавания динамики ТП. Такой подход отражает современную парадигму программного обеспечения – Software 2.0, которая, в отличие от Software 1.0, не предполагает явное написание кода алгоритма, а предусматривает создание нейронной сети определенной архитектуры, которая уже сама обучается (настраивается) на решение конкретной прикладной задачи.

Глубокая нейронная сеть моделирует иерархические абстракции в данных, используя архитектуры, состоящие из каскадного множества нелинейных преобразований (фильтров). На сегодняшний день существует несколько популярных архитектур *DNN*: неокогнитрон, автокодировщики, сверточные нейронные сети (convolutional neural network, CNN), ограниченная машина Больцмана, глубокие сети доверия, сети долго-краткосрочной памяти, управляемые рекуррентные нейронные сети, сети остаточного обучения [Созыкин, 2017]. В дальнейшем будут использоваться сверточные нейронные сети.

Архитектура нейронной сети определяет пространство гипотез, то есть количество классов, на которые разбиваются наборы входных данных. Предлагаемая архитектура глубокой нейронной сети для отслеживания динамики объекта представлена на рисунке 1. Она использует удачные практики ансамблевого применения нейронных сетей [Frazão et al., 2014], [Koitka et al., 2017], [Puchkov et al., 2018], но отличается от них наличием блока временной задержки, сигнал с которого также подается на выходной каскад сети для возможности распознавания изменений состояния ТП.

После расчета интервалов дискретизации по времени $\Delta t(i)$ для всех каналов, из них выбирается наименьший $\Delta t = \min_i (\Delta t(i))$ с целью дальнейшей синхронизации и унификации преобразований видеоданных, выполняемых нейронной сетью.

Также, для унификации дальнейших преобразований, входные каналные изображения с разрешениями $n[ip] \times m[ip]$ пикселей, перед подачей на вход нейронной сети, нормируются к одному размеру $n \times m$

пикселей, меньшему, чем наименьший из $n[ip] \times m[ip]$.

В блоке временной задержки происходит сдвиг на Δt момента получения изображения, что обеспечивает возможность расчета дискретных аналогов производных при определении изменений значений на выходах нейронной сети.

Входной каскад глубокой нейронной сети содержит несколько, параллельно работающих CNN, классифицирующих изображения с видеокамер соответствующего информационного канала, взятые через интервалы Δt . Эта процедура заключается в формировании в каждый момент времени Δt канального выходного вектора $V(j|i), j=1, 2, \dots, cl(i)$, где $cl(i)$ – количество классов для i -го информационного канала.

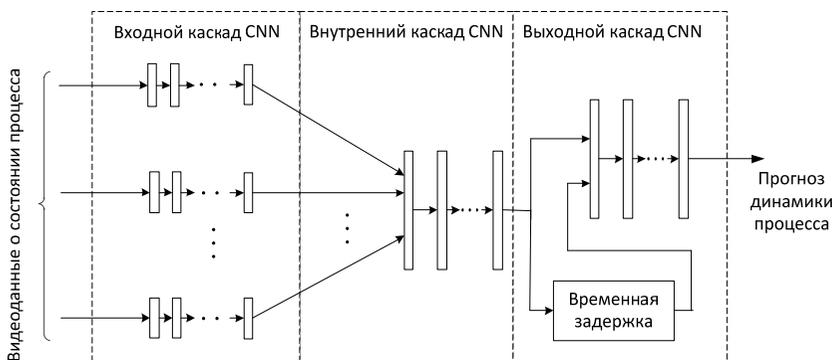


Рис. 1 Архитектура глубокой нейронной сети

CNN входного каскада формируют элементы векторов $V(j|i)$, которые принимают значения в диапазоне значений от 0 до 1, что отражает степень уверенности i -й канальной CNN в принадлежности контролируемого по изображению параметра к тому или иному классу в момент $t(k) = k\Delta t$, где k – порядковый номер отсчета времени.

Весь промежуток времени, в течении которого осуществляется ТП, разбивается на фрагменты длительностью ΔT . Будем считать, что они для всех информационных каналов приняты одинаковыми и определяются требованиями к периодичности поступления информации в АСУТП характеристиками самого критичного по частоте канала.

К моменту времени $T(\zeta) = \zeta\Delta T, \zeta = 1, 2, \dots, \psi$, где ψ – количество фрагментов длительностью ΔT , для i -го канала можно сформировать матрицу результатов классификации:

$$MV(i|\xi) = \begin{pmatrix} V(1|i,1) & V(1|i,2) & \dots & V(1|i,k) \\ V(2|i,1) & V(2|i,2) & \dots & V(2|i,k) \\ \dots & \dots & \dots & \dots \\ V(cl(i)|i,1) & V(cl(i)|i,2) & \dots & V(cl(i)|i,k) \end{pmatrix}, \quad (1)$$

где элемент $V(j|i,k)$, означает j -й выход CNN входного каскада i -го информационного канала в k -й отсчет времени $k\Delta t$.

Матрица (1), по сути, отражает то, как изменяется уверенность CNN в результатах классификации при прохождении ТП интервала ΔT под номером ζ .

На внутренний каскад CNN поступает тензор, состоящий из объединения частных матриц (1) для всех информационных каналов. На выходе внутреннего каскада образуется матрица MS , аналог матрицы (1), но содержащий уже большее количество элементов в соответствии с числом информационных каналов и классов состояний ТП по каждому каналу.

Для представления о динамике всего ТП рассчитывается тензор DV , содержащий отношения приращений элементов матрицы MS к интервалу дискретизации Δt . Ранг этого тензора будет равен трем, а его ζ -й срез имеет вид:

$$DV(\xi) = \begin{pmatrix} \frac{MS_{\xi}(1,1) - MS_{\xi-1}(1,1)}{\Delta t} & \dots & \frac{MS_{\xi}(1,I) - MS_{\xi-1}(1,I)}{\Delta t} \\ \frac{MS_{\xi}(2,1) - MS_{\xi-1}(2,1)}{\Delta t} & \dots & \frac{MS_{\xi}(2,I) - MS_{\xi-1}(2,I)}{\Delta t} \\ \dots & \dots & \dots \end{pmatrix}. \quad (2)$$

Срез (2) можно трактовать как изображение, поступающее на выходной каскад CNN. Смысловую нагрузку элементов среза (2) можно сопоставить с аналогом производных для непрерывных функций, так как они отражают изменение уверенности нейронной сети в принадлежности образа к тому или иному классу. Скорость изменения можно использовать для прогноза развития ТП.

Следует отметить, что количество классов для разных информационных каналов и разных технологических зон может быть различно. Поэтому, для обеспечения соразмерности, при подаче тензор DV на обработку нейронной сетью, некоторые срезы будут содержать нули на месте избыточных классов.

Тензор DV формировался на основе данных об изменениях уверенности CNN в классовой принадлежности параметров

информационных каналов, отраженной в матрице (1). Это позволяет предположить, что при формировании пространства гипотез можно ожидать хороших результатов прогноза с помощью предложенной архитектуры глубокой нейронной сети.

2. Результаты

Рекомендуемый выбор архитектуры глубокой нейронной сети предполагает обеспечение соответствия количества каналов поступления видеoinформации, количеству сверточных сетей во входном каскаде. В данном случае приведем результаты модельного эксперимента по обработке изображений слитков алюминиевого сплава с целью прогноза времени его полного расплавления. Процесс плавки занимает примерно 300 секунд, агрегатное состояние оценивается по изображению поверхности, наблюдаемой через смотровое окно, расположенное в крышке плавильной печи [Шкундин и др., 2018].

Разработка программы модели осуществлялась в IDE Spyder из сборки Anaconda (версия для операционной системы Linux) на языке Python 3.6. CNN создавались с использованием специализированной нейросетевой библиотеки Keras, являющейся надстройкой над фреймворком тензорных вычислений TensorFlow [Жерон, 2018]. Для визуализации процесса работы TensorFlow применялся фреймворк TensorBoard.

В рассматриваемом примере используется только один информационный канал – от технологической зоны плавления, поэтому структура применяемой сети значительно упрощена. Сеть содержит семь чередующихся слоев свертки и подвыборки и один выходной полносвязный слой с четырьмя выходами – по количеству распознаваемых классов агрегатного состояния вещества: класс «твердое» (0 – 269 сек.), «начальное переходное» (270 – 279 сек.), «конечное переходное» (280 – 289 сек.), «жидкое» (290 – 300 сек.). Таким образом, с классом связан временной диапазон, поэтому при классификации будет прогнозироваться время наступления того или иного агрегатного состояния.

На рисунке 2, а показаны изображения зоны плавления, взятые в разные моменты времени, на рис. 2, б – изображение тензора DV , отражающее динамику изображений на рис. 2, а. Последовательность изображений 2, б образует тренд динамики процесса плавления, который распознается CNN.

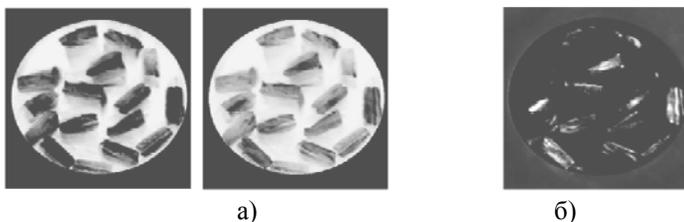


Рис. 2. Обрабатываемые изображения

Исходная обучающая выборка для каждого класса содержала 1600 примеров, тестирующая – 400 примеров. Для снижения возможности переобучения сети, помимо стандартно используемого в CNN метода Dropout, применялась аугментация – для исходных изображений рабочей поверхности слитков реализовывались сдвиги, изменение масштаба, повороты относительно центра, зеркальное отражение, аффинные преобразования. В результате этих процедур общий размер обучающей выборки составил 32000 примеров, тестирующей – 8000 примеров. Тренировка сети проводилась в течении 110 эпох. Функцией потерь для CNN была задана категориальная энтропия.

Разбиение видеоданных на кадры с конвертацией в формат jpg осуществлялось с помощью программной утилиты Free Video to JPG Converter с заданной дискретностью одна секунда. Такой подход позволили получить достаточный объем обучающей выборки для CNN, так как даже из видеозаписи одного процесса плавки можно выделить большое количество изображений.

Обучение проводилось на видеокарте GeForce GTX 1060, установленной на ноутбуке Asus FX502VM с CPU IntelCore i7-7700HQ, что обеспечило более чем двадцатикратный выигрыш по времени по сравнению с обучением на штатном процессоре ноутбука. Качество обучения CNN было отражено в метрике точности (accuracy), которая на тестовой выборке составила 77 %. Графики поведения функций потерь (loss) и точности показаны на рис.3.

Матрица неточностей (Confusion Matrix, CM), используемая для оценки качества классификации, в данном случае будет иметь четыре строки и четыре столбца. Ее столбцы отражают фактические данные, а строки – результаты работы классификатора.

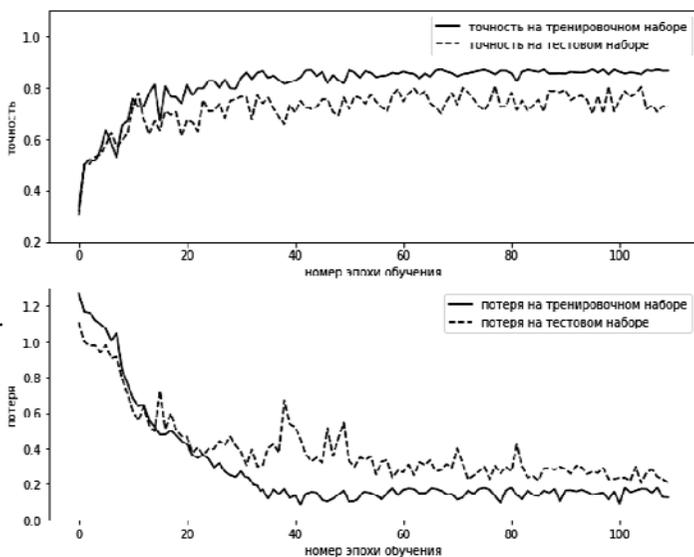


Рис. 3. Графиков поведения функции потерь и точности

При заполнении матрицы увеличивается на единицу число, стоящее на пересечении строки класса, который вернул классификатор и столбца класса, к которому действительно относится объект. В рассматриваемом примере каждый временной интервал, соответствующий разным стадиям плавления, был разбит на 10 подинтервалов ΔT . После проведенных экспериментов было осуществлено заполнение матрица CM :

$$CM = \begin{pmatrix} 7 & 2 & 1 & 0 \\ 1 & 6 & 2 & 1 \\ 2 & 1 & 6 & 1 \\ 0 & 2 & 1 & 7 \end{pmatrix}$$

Анализ значений элементов матрицы CM показывает, что большинство классов распознаны верно, так как диагональные элементы матрицы явно выражены.

Заключение

В процессе проведенного исследования возможности применения глубоких (сверточных) нейронных сетей для отслеживания динамики

состояния технологических объектов получены следующие результаты:

1. Предложена архитектура глубокой нейронной сети для получения информации о динамике состояния технологического объекта на основании видеоданных, поступающих из различных технологических зон. В своей основе она содержит сверточные нейронные сети с дифференциацией каскадов на входной, внутренней и выходной, что позволяет интегрировать различные информационные потоки в единый набор и повышать уровень абстракции представления данных. В выходном каскаде глубокой нейронной сети используется временная задержка в обработке изображений для обеспечения возможности распознавания динамики состояния объекта и ее прогноза.

2. Представлены рекомендации по выбору и адаптации архитектуры нейронной сети в зависимости от количества технологических зон и числа информационных видеоканалов.

3. Приведены результаты модельного эксперимента, которые показывают работоспособность предложенной архитектуры глубокой нейронной сети, что может свидетельствовать о целесообразности ее использования в различных прикладных областях, где необходим контроль динамики протекающих процессов по имеющейся видеоинформации.

Список литературы

[Ahlawat et al., 2019] Ahlawat S., Batra V., Banerjee S., Saha J., Garg A.K. (2019) Hand Gesture Recognition Using Convolutional Neural Network. In: Bhattacharyya S., Hassanien A., Gupta D., Khanna A., Pan I. (eds) International Conference on Innovative Computing and Communications. Lecture Notes in Networks and Systems, vol 56. Springer, Singapore.

[Cha 2017] Cha Y.J., Choi W. (2017) Vision-Based Concrete Crack Detection Using a Convolutional Neural Network. In: Caicedo J., Pakzad S. (eds) Dynamics of Civil Structures, Volume 2. Conference Proceedings of the Society for Experimental Mechanics Series. Springer, Cham.

[Fan et al., 2018] Fan Y., Lam J.C.K., Li V.O.K. (2018) Multi-region Ensemble Convolutional Neural Network for Facial Expression Recognition. In: Kůrková V., Manolopoulos Y., Hammer B., Iliadis L., Maglogiannis I. (eds) Artificial Neural Networks and Machine Learning – ICANN 2018. ICANN 2018. Lecture Notes in Computer Science, vol 11139. Springer, Cham.

[Frazão et al., 2014] Frazão X., Alexandre L.A. (2014) Weighted Convolutional Neural Network Ensemble. In: Bayro-Corrochano E., Hancock E. (eds) Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP 2014. Lecture Notes in Computer Science, vol 8827. Springer, Cham.

[Kasnesis et al., 2018] Kasnesis P., Patrikakis C.Z., Venieris I.S. (2019) PerceptionNet: A Deep Convolutional Neural Network for Late Sensor Fusion. In: Arai

К., Kapoor S., Bhatia R. (eds) Intelligent Systems and Applications. IntelliSys 2018. Advances in Intelligent Systems and Computing, vol 868. Springer, Cham.

[**Koitka et al., 2017**] Koitka S., Friedrich C.M. (2017) Optimized Convolutional Neural Network Ensembles for Medical Subfigure Classification. In: Jones G. et al. (eds) Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2017. Lecture Notes in Computer Science, vol 10456. Springer, Cham.

[**Kori et al., 2018**] Kori A., Soni M., Pranjali B., Khened M., Alex V., Krishnamurthi G. (2019) Ensemble of Fully Convolutional Neural Network for Brain Tumor Segmentation from Magnetic Resonance Images. In: Crimi A., Bakas S., Kuijff H., Keyvan F., Reyes M., van Walsum T. (eds) Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. BrainLes 2018. Lecture Notes in Computer Science, vol 11384. Springer, Cham.

[**Li et al., 2016**] Li H., Qian X., Li W. (2017) Image Semantic Segmentation Based on Fully Convolutional Neural Network and CRF. In: Yuan H., Geng J., Bian F. (eds) Geo-Spatial Knowledge and Intelligence. GRMSE 2016. Communications in Computer and Information Science, vol 698. Springer, Singapore.

[**Puchkov et al., 2018**] Puchkov A., Dli M., Kireyenkova M. (2020) Fuzzy Classification on the Base of Convolutional Neural Networks. In: Hu Z., Petoukhov S., He M. (eds) Advances in Artificial Systems for Medicine and Education II. AIMEE2018 2018. Advances in Intelligent Systems and Computing, vol 902. Springer, Cham.

[**Thielicke 2019**] Thielicke W. (2019). PIVlab - particle image velocimetry (PIV) tool(<https://www.mathworks.com/matlabcentral/fileexchange/27659-pivlab-particle-image-velocimetry-piv-tool>), MATLAB Central File Exchange. Retrieved April 24, 2019.

[**Xiang 2016**] Xiang, L., et al.: Automatic vehicle identification in coating production line based on computer vision. In: International Conference on Computer Science and Engineering Technology, pp. 260–267. World Scientific Publication Co. Pvt. Ltd. (2016).

[**Жерон 2018**] Жерон О. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow: концепции, инструменты и техники для создания интеллектуальных систем./ М.: «Диалектика», 2018. - 688 с.

[**Нагиев и др., 2017**] Нагиев А. Г., Садыхов В. В., Нагиев Г. А. Проблема апертурной задержки в цифровых системах измерения и её аналитическое решение методом матричной экспоненты//«Измерительная техника», №9, 2017. С.16-20.

[**Похабов 2019**] Похабов Ю.П. Проблемы надёжности и пути их решения при создании уникальных высокоответственных систем. Надежность. 2019; 19(1):10-17. <https://doi.org/10.21683/1729-2646-2019-19-1-10-17>

[**Рубио 2016**] Рубио Д. Видеоаналитика: возможности и решения [Текст] / Д. Рубио // Современные технологии автоматизации (СТА). - 2016. - № 4. - С. 86-92.

[**Сакович и др., 2014**] Сакович И.О., Белов Ю.С. Применение метода Лукаса — Канаде для вычисления оптического потока. Инженерный журнал: наука и инновации, 2014, вып. 7. URL: <http://engjournal.ru/catalog/pribor/optica/1275.html>

[Созыкин 2017] Созыкин А. В., Обзор методов обучения глубоких нейронных сетей, Вестн. ЮУрГУ. Сер. Выч. матем. информ., 2017, т. 6, выпуск 3, 28–59 с.

[Шкундин и др., 2018] Шкундин С.З., Колистратов М.В., Белобокова Ю.А. Проверка работоспособности алгоритмов определения изменения агрегатного состояния металла // Системный администратор.– 2018. – №10 (191) – С. 90-93.

[Шолле 2018] Шолле Ф. Глубокое обучение на Python.– SPb.: Peter, 2018.– 400 с.

[Шунина 2015] Шунина Ю. С., Алексеева В. А., Клячкин В. Н. Критерии качества работы классификаторов // Вестник УлГТУ. 2015. №2 (70). URL: <https://cyberleninka.ru/article/n/kriterii-kachestva-raboty-klassifikatorov>

АНАЛИЗ РАДИОЛОКАЦИОННЫХ ДАННЫХ С ПОМОЩЬЮ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ЗАДАЧИ КЛАССИФИКАЦИИ АГЕНТОВ ТРЕХМЕРНОЙ ДОРОЖНОЙ СЦЕНЫ

А.В. Аверин (*averin-artem2007@yandex.ru*),
НИТУ МИСиС, Москва

И.А. Костин (*kostin.ivan.a@gmail.com*),
НИТУ МИСиС, Москва

М.Ю. Мазуров (*mazurov.myu@phystech.edu*),
НИУ МФТИ, Долгопрудный

Н.В. Панокин (*n.panokin@misis.ru*),
НИТУ МИСиС, Москва

В работе описывается архитектура модели реализующая сенсорный и имитационный подходы к получению и обработке радиолокационных данных. Описаны протяженные радиолокационные цели, используемые в модели трехмерной дорожной сцены. Приведена архитектура сверточной нейронной сети для задачи классификации агентов трехмерной дорожной сцены и результаты работы алгоритмов оценки геометрических размеров радиолокационных объектов.

Ключевые слова: архитектура, модель, датчик, классификация, сверточная нейронная сеть, нейросетевой дескриптор, DBSCAN.

Введение

Проблема распознавания дорожных агентов в рамках задачи анализа трехмерной дорожной сцены представляет повышенный интерес в связи с развитием беспилотного транспорта. Повышение степени автоматизации транспортных средств, роботизированных комплексов и платформ требует высоких результатов классификации объектов, предъявляемых к алгоритмам принятия решений. Датчики, построенные на разных физических принципах (радиолокаторы, лидары, видеокамеры, акустические датчики и т.п.), в той или иной степени позволяют проводить анализ трехмерных дорожных сцен. Преимущества и недостатки применяемых основных датчиков приведены в рисунке 1.

	GPS/INS	Camera	LIDAR	HD Radar	LGPR
Urban	Dark	Light	Dark	Dark	Dark
Rain	Light	Dark	Dark	Dark	Dark
Dust/Fog	Dark	Light	Dark	Dark	Dark
Snow	Dark	Light	Dark	Dark	Dark
Tunnels	Dark	Dark	Dark	Dark	Dark
Contested Spectrum (Blind, jam, interfere)	Dark	Dark	Light	Dark	Light
Subsurface Changes	Dark	Dark	Dark	Dark	Light
Surface Changes	Dark	Light	Dark	Dark	Dark

Рис. 1. Сравнение ограничений основных видов используемых датчиков

Область применения основных датчиков на беспилотных транспортных средствах ограничена прежде всего несколькими существенными факторами: для видеокамер – это значительное снижение качества изображения в сложных погодных условиях (сильный снегопад, дождь, туман), для лидаров – нестабильная работа в сложных погодных условиях в совокупности с высокой стоимостью, для акустических датчиков – небольшая дальность работы. Автомобильные радары обладают определенными преимуществами в сравнении с вышеперечисленными датчиками. На работу автомобильного радара, работающего в определенных диапазонах длин волн (в так называемых окнах прозрачности), не оказывают влияния погодные условия и «засветы» от фар встречных и попутных машин, а также солнечных лучей. Невысокая стоимость изделия и высокое качество разрешения объектов позволяет рассматривать радар как один из основных источников данных при комплексировании информации с разнородных физических датчиков. Вместе с тем, для автомобильных радарных систем необходимо адаптировать известные алгоритмы первичной и вторичной обработки радиолокационной информации вследствие специфики использования. Для отработки методов и интерпретации трехмерных данных с целью повышения эффективности детектирования и распознавания объектов необходимо создавать реалистичные модели трехмерных дорожных сцен.

Существуют два подхода к получению радиолокационных «портретов» трехмерных дорожных сцен. Сенсорный подход основан на анализе данных с камер и лидаров, которые формируют систему технического зрения. Данные, получаемые с этих систем, можно промоделировать в существующих средах имитационного моделирования AirSim, CARLA, V-Rep, Gazebo. Вариант сенсорного подхода получения радиолокационных

данных [Беляев, 2018] в среде CARLA (Car Learning to Act) позволяет создавать изображения дорожной сцены, а также семантическую сегментацию и карту глубины (рисунок 2). Объекты дорожной сцены представляются набором блестящих точек. Построение радиолокационных портретов дорожных трехмерных объектов производится путем анализа суперпозиции принимаемых сигналов. Однако, стоит учитывать, что протяженные объекты, коими можно считать дорожные препятствия, мосты, надземные пешеходные переходы и крупногабаритные транспортные средства [Гоноровский, 1954], в силу своих геометрических размеров превышают длину элемента разрешения по дальности при облучении их в диапазоне миллиметровом диапазоне длин волн. Одним из вариантов решения задачи рассеивания на модельных объектах трехмерной дорожной сцены является представление сложных объектов в виде группы точечных отражателей, распределенных по его поверхности. В миллиметровом диапазоне длин волн количество центров рассеивания возрастает и на модель все большее влияние оказывает флуктуационный характер изменения интенсивности отраженного сигнала по причине случайных фазовых набегов на объектах, размеры которых сравнимы с длиной волны.



Рис.2. Среда моделирования CARLA

Имитационный подход получения радиолокационных данных вплоть до частот 79 ГГц и выше реализован в среде WaveFareg [Электронный ресурс, 2015], функционал которой позволяет решать следующие задачи:

- изменять тип антенной системы, учитывать многолучевое распространение и атмосферные эффекты при распространении сигнала;
- формировать объекты дорожной сцены с помощью встроенного редактора или загружать существующие нарисованные трехмерные объекты;
- получать радиолокационные портреты дорожных объектов на плоскости скорость-дальность;
- оптимизировать места установки автомобильного радара с целью повышения качества обработки сигнала и повышения классификаций целей;
- проводить быстрый анализ повторяющихся сценариев дорожной сцены;
- использовать сторонние программные продукты для последующего анализа принимаемых сигналов.

В свою очередь, имитационный способ получения радиолокационной информации часто является узкоспециализированным и не всегда позволяет в полной мере исследовать качество методов детектирования и распознавания объектов трехмерной дорожной сцены в реальных условиях быстроменяющейся дорожной обстановки. Вместе с тем, на рынке отсутствуют среды моделирования радиолокационных систем ближнего (Short Range Radar - SRR) и дальнего (Long Range Radar – LRR) действий [Беляев, 2018], которые сочетали бы возможность имитационного и сенсорного подходов получения радиолокационных данных с возможностью обработки методов детектирования и распознавания трехмерных объектов, и вместе с тем имели модульную структуру, позволяющая быстро и гибко прототипировать различные варианты решения радиолокационных задач. Кроме того, в анализе методов обработки радиолокационной информации необходимо наличие комплексной системы, позволяющей применять оба этих подхода.

Таким образом, существует необходимость разработки методов математического, в том числе, имитационного моделирования с целью выяснения эффективности разрабатываемых методов обработки радиолокационной информации и уточнения требуемых характеристик 3D радара, используемого для построения трехмерной дорожной сцены. Функционал данной модели включает в себя:

- частотное качание луча в плоскости угла места;
- возможность моделирования радара в диапазонах 24 и 77 ГГц;

- возможность моделирования объектов дорожной сцены;
- учет амплитудно-частотных рассогласований в приемных каналах;
- модуль визуализации результатов обработки радиолокационного сигнала;
- учет влияния подстилающей поверхности;
- модуль классификации радиолокационных данных в соответствии с принятыми априорными сведениями о параметрах дорожных агентов.

В данной статье рассматривается верхняя архитектура модели, сочетающая имитационный и сенсорный подходы, с помощью которой могут быть синтезированы модельные данные распределённых объектов, также реальные радиолокационные данные с дорожной обстановкой. также описан нейросетевой подход к задаче классификации агентов трехмерной дорожной сцены.

1. Архитектура программного обеспечения для формирования трехмерной дорожной сцены

Архитектура программного обеспечения (ПО) для обработки радиолокационной информации в случае сигнала с линейной частотной модуляцией и формирования трехмерной дорожной сцены разработана согласно основным принципам проектирования [Гамма, 2001] и отражает основные требования верхнего уровня ко всем составляющим частям системы обработки. Программный компонент, реализующий формирование дорожной сцены на основе радиолокационных сигналов, должен осуществлять:

- a. управление периферийными устройствами;
- b. считывание и синхронизацию данных с радиолокатора, видеокамеры, CAN-шины;
- c. обработку данных в одном из выбранных оператором режимов:
 - обработка моделированных радиолокационных данных;
 - обработка ранее записанных экспериментальных данных;
 - обработка данных в режиме реального времени;
- d. визуализацию результатов обработки данных.

Общая структура ПО представлена на рисунке 3 и состоит из:

1. модуля дорожной сцены;
2. модуля формирования сигналов биений;
3. диспетчера (синхронизатора);
4. модуля автокалибровки;
5. модуля первичной обработки;

6. модуля вторичной обработки;
7. модуля визуализации.

Архитектура системы предполагает обработку данных, полученных из различных источников. Сигналы биений могут быть получены непосредственно от радара и обработаны сразу, либо сначала записаны на информационный носитель системы хранения данных (например, жесткий диск), а затем прочитаны с него для обработки, либо получены от модуля формирования сигналов биений. В первых двух случаях, при этом, будут обрабатываться реальные данные, а в последнем – смоделированные.

2. Типы распределенных объектов, наполняющих трехмерную дорожную сцену

Распределенные объекты характеризуются следующими параметрами: вектором начального положения; вектором скорости; значением ЭПР.

Для дорожного ограждения дополнительно задается высота и длина ограждения, скважность (расстояние между соседними элементами конструкции).

На приведенных далее рисунках представлены основные распределенные объекты дорожной сцены: легковой и грузовой автомобили, мотоцикл (рисунок 5) и столбы, а также отбойники и металлические заборы (рисунок 6).

Модуль имитации дорожной сцены предусматривает расширение списка моделируемых объектов путем добавления новых систем блестящих точек и написания программных функций, описывающих законы движения новых объектов (рисунок 7).

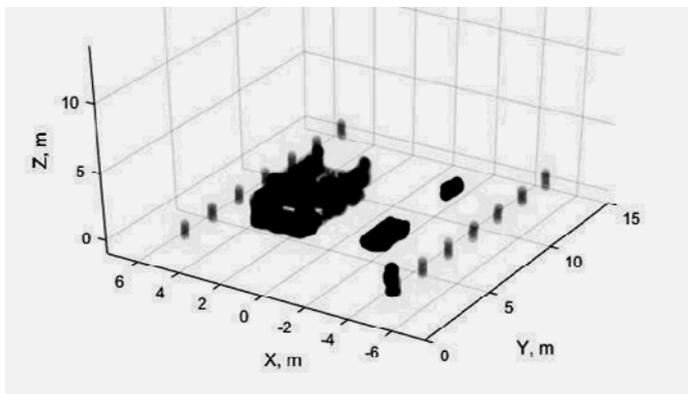


Рис.6. Пример моделируемой трехмерной дорожной сцены (отбойники по краям дороги, человек у края проезжей части, грузовой автомобиль, легковой автомобиль, мотоцикл.)

3. Модуль классификации и кластеризации протяженных радиолокационных объектов

Кластеризация отметок необходима для определения принадлежности той или иной отметки к конкретному объекту трехмерной дорожной сцены и объединения отметок в облака точек. Помимо эвристических методов, основанных на определении меры расстояния между отметками (аналогично методу имитации отжига [Кирсанов, 2007]), применяется широко распространенный метод k-средних [Coates, 2012], суть которого заключается в минимизации квадратичного отклонения отметок от центров соответствующих им объектов. Соответственно, минимизируемая функция выглядит следующим образом:

$$V = \sum_{i=1}^M \sum_{x \in C_i} \left\| \hat{x} - \hat{\mu}_i \right\|^2,$$

где \hat{x} – 4-хмерный вектор, соответствующий каждой отметке, а $\hat{\mu}_i$ – «центр масс» соответствующего кластера C_i . Основным недостатком данного метода является отсутствие гарантии сходимости к решению и сильная зависимость от начального выбора центров кластеров. В связи с этим, указанный метод используется со следующими модификациями:

- предварительный анализ отметок с целью определения количества кластеров;
- задание положений центров кластеров на основе

предварительной обработки множества отметок;

- оптимизация нормы $\left\| \hat{x} - \hat{\mu}_i \right\|$ в пространстве \mathbf{R}^6

взвешиванием.

Результаты применения данного метода выражаются в объединении близкорасположенных отметок со сходными характеристиками в группы и изображены на рисунке 7.



Рис.7. Кластеры отражающих точек с наложением на видеозображение (цвет отображает расстояние до объекта)

Квалификация может проводиться как для отдельных отметок, так и для выделенных кластеров. Целью квалификации является определение типа объекта, выделение препятствий, дорожных конструкций, других транспортных средств и иных участников дорожного движения. В классической теории радиолокации и распознавании образов квалификация целей базируется на априорной информации о количестве и качественном составе классов объектов, причем известно, что с ростом числа классов ее достоверность падает [Горелик, 1963]. Набором признаков в данном случае является вектор $\hat{x}_e = [R, V, \theta, \varphi, l, a]$ в \mathbf{R}^6 , где R – радиальная дальность, V – радиальная скорость, θ – направление по азимуту, φ – направление по углу места, l – оценка линейного размера, a – ЭПР. Однако, в настоящее время широко распространены технологии машинного и глубокого обучения для обнаружения и квалификации объектов трехмерной дорожной сцены, где в качестве исходных данных выступают характеристики массивов отражающих точек. Для использования данных методов существует ряд предпосылок:

- Массивы, формируемые алгоритмами пространственно-временной обработки радиолокационного сигнала, имеют структуру,

схожую с обычными изображениями, отличаясь от последних дополнительным измерением. Это обстоятельство дает возможность использовать методы и средства обработки изображений к радиолокационным данным после некоторых доработок.

- Методы и средства машинного обучения хорошо развиты и широко распространены в системах компьютерного зрения на базе видеокамер и лидаров.

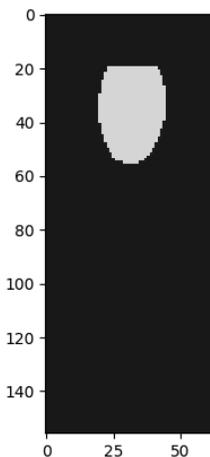
- Дополнительные параметры, такие как скорость и расстояние до объекта, позволяют повысить надежность используемых методов глубокого и машинного обучения, а также снять некоторые ограничения, актуальные для обработки видеоизображения.

Одним из таких методов автоматического анализа трехмерной дорожной сцены по радиолокационному изображению является подход, основанный на сверточных нейронных сетях [Wang, 2003], для предоставления информации о количестве и расположении значимых объектов на трехмерной дорожной сцене.

В решении этой задачи выделяется два этапа. Первый этап связан с тренировкой детектора для классификации части сцены и определения этой части, как некоторого объекта реального мира. Для тренировки такого детектора используется последовательная модель, состоящая из 3D-сверточных, объединяющих и полносвязных слоев. Подобная структура встречается в многих научных статьях для классификации 3D-геометрий. Используемая тренировочная выборка представляет собой набор участков радиолокационного изображения, размеченный по принадлежности к классу. Далее при подборе оптимальных гиперпараметров проводится обучение нейронной сети, максимизирующая вероятность правильного распознавания классов [Charles, 2016, Frome, 2004].

Второй этап заключается в прохождении этим детектирующим окном по всей сцене, тем самым предоставляя информацию о нахождении в областях сцены объектов определенных классов, данный подход используется в RCNN. Таким образом получается разреженная 3D-карта с областями кучности, где найти определенный класс можно с наибольшей вероятностью (рисунок 8). Найдя центры этих кучностей, можно получить информацию о геометрическом центре объекта данного класса, а выделив границы - оценить размеры объекта. Остановимся на приведенных этапах подробнее.

Рис.8. Выделение наиболее вероятных областей нахождения объекта при глубоком обучении в осях «дальность-азимут»



4. Постановка задачи классификации объектов

Существует трехмерная дорожная сцена, полученная путем первичной и вторичной обработок радиолокационных сигналов автомобильного радиолокатора. В области, формирующей сцену, могут находиться разнообразные дорожные агенты. Задача детектирования объектов необходима для поиска расположения объектов и определения их геометрических размеров. Задача детектирования объектов сводится к анализу 4-х мерного облака точек, где первые три параметра облака точек – пространственные координаты, четвертый параметр – энергетический, описывающий мощность принимаемого сигнала от конкретной блестящей точки. Нейросетевой подход к анализу видеоданных возможно адаптировать для 3-х мерных объектов путем использования алгоритмов анализа воксельных данных.

Так как задача сходна с принципами нахождения локальных пространственных черт (local perception field) в пространстве признаков, необходимо использовать сверточные нейронные сети с трехмерными слоями для анализа трехмерных данных. При работе с радиолокационными изображениями низкой детализации нет необходимости в использовании большого числа сверточных слоев. Выходы сверточных слоев объединяются в классификационную часть, которая отвечает за определение класса объекта, и регрессионную часть, к которой подведены выходы с результатами классификации (рисунок 9).

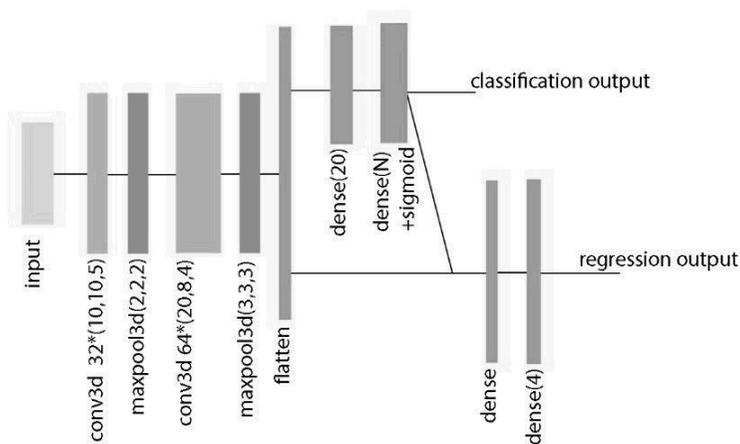


Рис.9. Функциональная схема нейросетевого дескриптора

5. Обучение нейронной сети и выбор алгоритма классификации

На вход модели поступает часть спектральной плотности мощности размером $[40 \times 20 \times 16]$ элементов, что продиктовано максимальными ограничениями на размеры характерных агентов дорожной сцены. Путем последовательного анализа части спектральной плотности мощности строится регрессионная карта размеров и карта вероятностей классов. Таким образом, предложенный вариант сети является аналогом R-CNN сети без алгоритма выборочного поиска (selective search), но модифицирована для работы с 3x-мерными данными. Наиболее простой подход к анализу геометрических центров и размеров интересующих объектов реализуется путем использования стандартных функций библиотеки компьютерного зрения OpenCV, позволяющих выделять контуры на срезе карт в плоскости «азимут-дальность». Результат работы данного подхода в силу возможной сильной интерференции между группой блестящих точек является неудовлетворительным. По контурам областей не представляется возможным разрешать объекты (рисунок 10).

Рис.10. Карта классификации для работы с OpenCV (слева) и пример карты регрессии на выходе из нейросетевого дескриптора (справа)

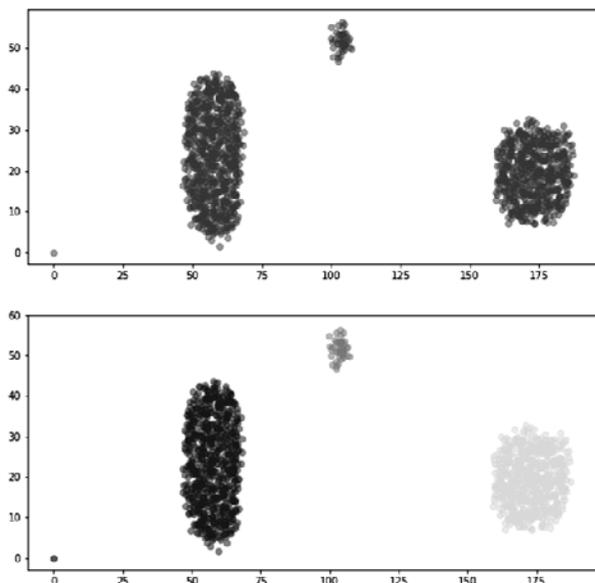
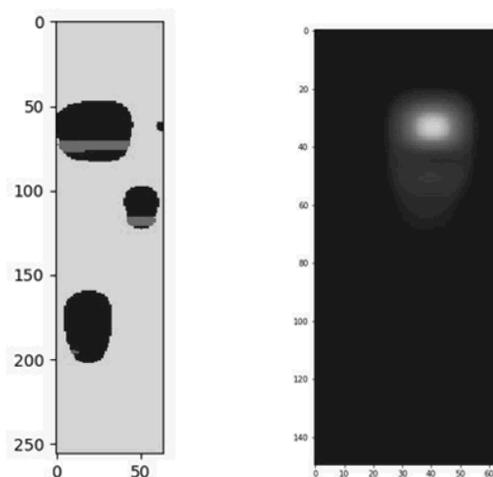


Рис.11. Пример работы алгоритма DBSCAN. На рисунке изображен срез облака точек для регрессионной карты (сверху). На втором изображении видим результат кластеризации (снизу).

Повышение точности кластеризации группы блестящих точек (рисунок 11) удается реализовать при помощи алгоритма DBSCAN [Ester, 1996]. Генерирование модельных данных теперь подчиняется заданному закону распределения вероятностей (рисунок 12). Варьируя основные параметры алгоритма, удастся повысить точность определения размеров областей.

Преимущество алгоритма DBSCAN в сравнении с популярным алгоритмом K-means заключается в отсутствии необходимой для последнего априорной информации о количестве кластеров.

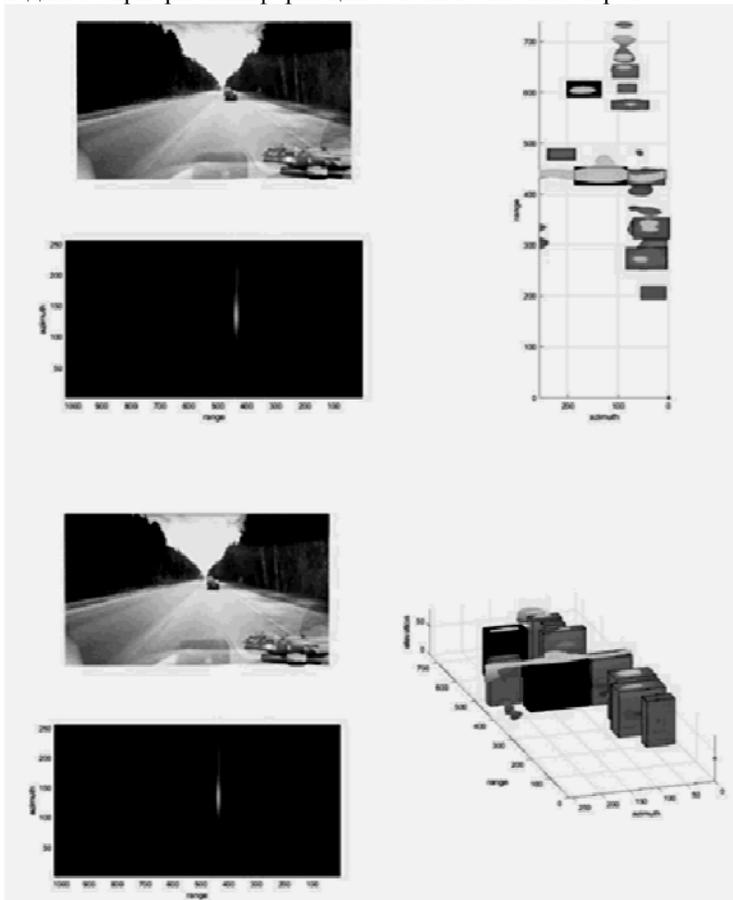


Рис. 12. Пример разметки радиолокационных данных, получаемых с реальной трехмерной дорожной сцены (вид сверху (сверху) и аксонометрическая проекция карты разметки (снизу))

После нахождения центра объекта на картах, размеры по осям получаются через обращение к регрессионным размерным картам. В итоге для каждого объекта получается область (bounding box), в которой лежат его границы, и класс объекта. Стоит отметить, что для нейросетевого дескриптора функции ошибок следующие: для классификации - перекрестная энтропия (categorical entropy), для регрессии - MSE. Для оценки алгоритма выделения объектов по классификационным картам, полученных с помощью нейросетевого дескриптора, используются аналогичные обычным 2-мерным изображениям функции ошибок, заключающиеся в сравнения истинного bounding box с предсказанным для каждого объекта (рисунок 12).

Заключение

В работе рассмотрена архитектура модели, сочетающая имитационный и сенсорный подходы к формированию радиолокационных данных для решения задачи классификации агентов дорожной трехмерной сцены. Показаны преимущества описанной модели в сравнении с вариантами реализующие описанные подходы по отдельности. Приведены типы распределенных радиолокационных объектов, позволяющих смоделировать реалистичную трехмерную дорожную сцену. Описана структура и принцип работы модуля классификации и кластеризации данных. Показаны подходы к классификации радиолокационных данных при помощи сверточных нейронных сетей, приведена архитектура нейронной сети, предложена реализация алгоритма DBSCAN для задачи кластеризации данных, показан вариант объемной разметки радиолокационных данных. Применяемая сверточная нейронная сеть адаптирована для анализа воксельных данных, представляющие собой области спектральной плотности мощности. Благодаря описанному методу обработки трехмерных радиолокационных изображений, возможно не только проводить оценку положения и геометрических размеров объектов дорожной сцены, но и их классификацию.

Список литературы

[Беляев, 2018] Инженерный вестник Дона, №2 (2018) Моделирование работы автомобильного радара в задаче автономного движения. А.А. Беляев, Т.А. Суанов, Д.О. Троц.

[Гамма, 2001] Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – СПб: Питер, 2001. – 368с.: ил. (Серия «Библиотека программиста»).

[Гоноровский, 1954] Гоноровский И.С. Радиосигналы и переходные явления в

радиоцепях. М.: 1954 г. - 326с.

[Горелик, 1963] А.Л. Горелик, Ю.Л. Барабаш. Селекция и распознавание на основе локационной информации. – М.: Радио и связь, 1990 – 240 с. || Ю.Л. Барабаш, Б.В. Варский, В.Т. Зиновьев. Автоматическое распознавание образов – Киев: КВАИУ, 1963 – 168 с.

[Кирсанов, 2007] Кирсанов М.Н. Графы в Maple. – М.: Физматлит, 2007. – С.151-154.

[Электронный ресурс] <https://www.remcom.com/wavefarer-automotive-radar-software>. (дата обращения 3.12.2015)

[Charles, 2016] Charles R. Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, Leonidas J. Guibas. Volumetric and Multi-View CNNs for Object Classification on 3D Data. in Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2016.

[Coates, 2012] Adam Coates and Andrew Y. Ng. Learning Feature Representations with K-means, Stanford University, 2012.

[Frome, 2004] A. Frome, D. Huber, and R. Kolluri, “Recognizing objects in range data using regional point descriptors,” ECCV, vol. 1, pp. 1–14, 2004.

[Wang, 2003] C.C. Wang, C. Thorpe, and S. Thrun, “Online Simultaneous Localization and Mapping with Detection and Tracking of Moving Objects: Theory and Results from a Ground Vehicle in Crowded Urban Areas,” in Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2003.

[Ester, 1996] Ester M., Kriegel H.P., Sander J., XiaoweiXu A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Published in Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96).

ПРИМЕНЕНИЕ МЕХАНИЗМА СИММЕТРИЙ В НЕЙРОННЫХ СЕТЯХ

А. Г. Ложкин (*lag@istu.ru*)
ИжГТУ имени М. Т. Калашникова, Ижевск

К. Н. Майоров (*gibiskus@gmail.com*)
ЗАО «ИННЦ», Ижевск
ИжГТУ имени М. Т. Калашникова, Ижевск

В статье рассмотрены основные функции активации в современных нейронных сетях и их недостатки. Сделан вывод, что все они обладают одним недостатком, который заключается в невозможности интерпретировать полученные сигналы, это просто нормализованные значения взвешенной суммы синапсов. Рассмотрена таблица симметрий (автоморфизмов) и их роль в семиотическом анализе и лингвистике. Сделано предположение о возможности использования механизма симметрий в нейронных сетях. Предложен новый подход, который включает в себя группировку нейронов в скрытом слое по виду симметрии (автоморфизма) и использовании трехфазовой функций активации для каждой группы, характеризующих проявление свойств автоморфизма данной группы. Предлагаемый подход может сделать нейронные сети более доступными для понимания ввиду интерпретируемости сигналов.

Ключевые слова: нейронные сети, функция активации, автоморфизмы, группы нейронов, формальные языки

Введение

Глубокие нейронные сети постепенно внедряются во все сферы человеческой деятельности. Это объясняет тот факт, что сеть может автоматически выделять из данных наиболее важные признаки для получения прогноза в отличие от других алгоритмов машинного обучения, где конструирование признаков ложится на плечи специалиста по работе с данными [Созыкин, 2017].

На сегодняшний день нам известно множество различных методов обучения нейронных сетей (градиентные, генетические, стохастические и др.). При этом качество обучения нейронной сети напрямую связано с

выбранной функцией активации нейронов, которая и придает нейронной сети столь выразительные возможности. Различия лишь в конструкции этой функции. При этом полученные значения активации нейронов нельзя как-то интерпретировать, кроме как степень возбужденности нейрона. Поэтому механизм принятия решений нейронной сетью остается сложным для понимания.

В данной статье рассматривается идея создания новой архитектуры нейронной сети с различными сложными функциями активации, которые бы могли отражать степень проявления свойств симметрий в поступающих на нейрон сигналах.

1. Функции активации нейронных сетей

Функция активации – это обычно нелинейная функция, принимающая на вход линейную комбинацию входных сигналов с их весами и вычисляющая выходной сигнал нейрона [Goodfellow et al., 2016]. По своей сути она приводит взвешенную сумму всех синапсов к нормализованному виду, который может являться как ответом всей сети, так и синапсом для нейронов следующего скрытого слоя. На рис.1 можно увидеть место функции активации в модели нейрона.

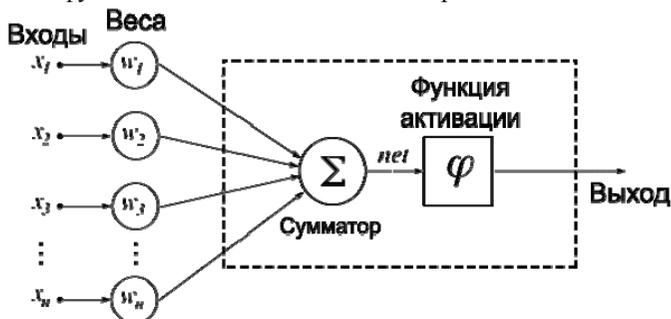


Рис.1.Схема математической модели нейрона

Ранее одной из самых распространенных нелинейных функций активации была функция логистического сигмоида (1.1).

$$f(x) = \frac{1}{1 + (e^{-\alpha x})}. \quad (1.1)$$

Параметр α описывает крутизну функции. Функция стремится к нулю при $x \rightarrow -\infty$ и единице при $x \rightarrow +\infty$.

В отличие от линейных и полулинейных функций функция логистического сигмоида показывает уровень активации нейрона и имеет гладкий градиент, что делает ее удобной для оптимизации градиентными

методами. Однако серьезным недостатком является перенасыщение функции в единице и ноль. При значениях линейной комбинации, близких к концам сигмоида, происходит эффект затухания градиента – дельта x практически не изменяется. Также выход функции не центрирован относительно нуля, что приводит к зигзагообразной динамике при изменении весов.

Еще один представитель семейства сигмоидных функций - функция гиперболического тангенса(1.2):

$$f(x) = th(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (1.2)$$

Функция очень похожа на функцию логистического сигмоида, но при $x \rightarrow -\infty$ функция стремится к -1 . В отличие от логистической функции функция гиперболического тангенса центрирована относительно нуля, что обеспечивает более быструю сходимость. При этом проблема затухающего градиента по-прежнему сохраняется – точками насыщения являются 1 и -1 .

Наиболее применяемая функция активации в современных нейронных сетях — функция линейного выпрямителя (1.3) или ReLu (rectified linear unit) [He et al., 2016a]. Такое название произошло из-за аналогии с однополупериодным выпрямителем в электротехнике. Функция не меняет сигнал, если он положительный, иначе принимает значение ноль.

$$f(x) = \max(0, x) \quad (1.3)$$

Функция линейного выпрямителя обладает несомненными преимуществами по сравнению с функциями сигмоида и гиперболического тангенса. Расчет производной не требует ресурсоемких операций, таких как вычисление экспоненты и возведение в степень — производная равна нулю при $x < 0$ и единице при $x > 0$. Это увеличивает скорость вычислений и придает нейронной сети большую выразительность, так как не перенасыщается сверху и показывает насколько сильно возбужден нейрон. Было даже зафиксировано 6-кратное увеличение скорости сходимости стохастического градиентного спуска по сравнению с функциями сигмоида и гиперболического тангенса [Krizhevsky et al., 2012].

При этом Relu не лишена недостатков. Изначально не активировавшиеся нейроны уже не смогут активироваться в процессе обучения из-за равенства нулю градиента ошибки. Высокий коэффициент скорости обучения может также привести к большой дельте ошибки и вследствие вечной дезактивации нейрона [Ramachandran et al., 2017]. Это означает, что результат напрямую зависит от правильно выбранной скорости обучения и инициализации весов в сети.

С проблемой вечной дезактивации нейронов справляются различные модифицированные версии ReLu. Например Leaky ReLU (1.4) отличается от обычного линейного выпрямителя тем, что на интервале $x < 0$ выход функции уже не равен нулю, а становится линейным с небольшим угловым коэффициентом (около 0,01). Это позволяет избежать появления нулевого градиента при отрицательных значениях сумматора и соответственно решает проблему полного отмирания ReLu-нейрона [Рудой, 2011].

$$f'(x) = \begin{cases} x, x > 0 \\ 0,01 * x, x \leq 0 \end{cases} \quad (1.4)$$

Parametric ReLu (1.5) - это улучшенная версия функции Leaky ReLu, в которой линейный коэффициент не задается по умолчанию, а подбирается для каждого нейрона градиентным методом [He et al., 2015b].

$$f'(x) = \begin{cases} x, x > 0 \\ \alpha x, x \leq 0 \end{cases} \quad (1.5)$$

При очень маленьком значении α функция становится идентичной LReLU.

Из рассмотренных функций активации наиболее актуальными сейчас являются ReLu и ее различные вариации [Xu et al., 2015]. Однако все они обладают одним недостатком – мы не можем как-то интерпретировать полученные сигналы, это просто нормализованные значения взвешенной суммы синапсов. Хотелось бы видеть выход нейрона как меру какого-то свойства. Решить эту проблему могло бы внедрение механизма симметрий.

2. Модель нейронной сети с механизмом симметрий

На основе семиотического анализа в интерпретации А.П. Ершова была получена таблица автоморфизмов (симметрий) евклидовой плоскости [Ложкин, 2010а]:

1. Существование множества ($A \neq \theta$ Цермело).
2. Существование отношения ($a_1 R a_2$ Кодл).
3. Принадлежность элемента множеству ($a \in A$ Фрэнкель).
4. Универсальное отношение ($f : \Omega \rightarrow \Omega'$ импликация).
5. Лингвистическое описание множества (Декарт).
6. Лингвистическое представление отношения (Декарт).
7. Сохранение кардинальности ($m(A) = const$ Лагранж).
8. Сохранение мощности отношения ($n = const$ in

$C_1x^n + C_2y^n + C_3x^{n-1}y^{n-1} + \dots + C_{k-1}x + C_ky + C_0$ Кляйн).

9. Лингвистический порядок ($\vec{v} = xi + yi + zk + w$ Гамильтон).

10. Математический порядок ($a_i < a_{i+1}$, где $a_i, a_{i+1} \in R$ Кантор).

11. Переставная симметрия ($a_i \leftrightarrow a_j$).

12. Зеркальная симметрия ($a_i \cdot -1 = -a_i$).

Идея нового подхода – сделать нейронную сеть более интерпретируемой и логичной за счет группировки нейронов в скрытом слое по виду симметрии (автоморфизма) из таблицы Дьедонне и использовании функций активации, характеризующих проявление свойств автоморфизмов каждой группы.

2.1. Архитектура нейронной сети на основе групп симметрий

Рассмотрим нейронную сеть, где скрытый слой состоит из подгрупп нейронов, относящихся к определенному типу симметрии. Размер групп нейронов в слое может задаваться вручную либо распределяться пропорционально количеству групп в слое. Схема такой нейронной сети

представлена на рис.2

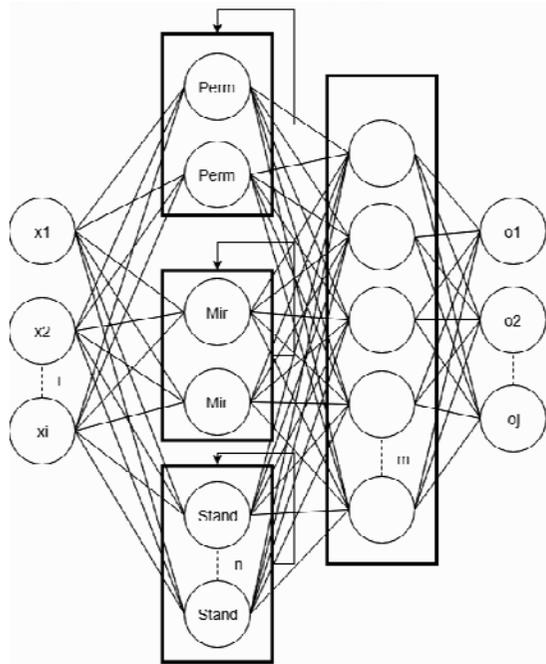


Рис.2. Схема нейронной сети с одним скрытым слоем на основе групп переставной и зеркальной симметрий.

У каждой группы нейронов предполагаются свои особенности: память и своя функция активации, базирующаяся на свойстве симметрии данной группы. Память нейрона представлена в виде таблицы (табл.1), состоящей из значений классической функции активации (Value) с их частотой (Frequence).

Табл.1

Index	Value	Frequence
1	0.311	5
2	0.653	2
n	0.427	10

Количество строк в такой таблице может задаваться вручную, по умолчанию планируется использовать 5.

На начальном этапе возьмем две группы симметрий – переставную и зеркальную, а также одну стандартную группу нейронов.

2.2. Трехшаговая функция активации.

Активация нейрона выполняется в три шага [Bozek al., 2018]:

Шаг 1. Получение значения классической функции активации.

На первом шаге значение взвешенной суммы нейрона передается в функцию гиперболического тангенса (1.2), так как для дальнейшей обработки нам нужен нормализованный сигнал, который также может принимать отрицательные значения.

Шаг 2. Генерация цепочки символов

На второй фазе мы генерируем цепочку символов, применяя теорию трансляции и формальных языков [Волкова, 2009].

Для генерации цепочки будем использовать три разряда после запятой от значения функции гиперболического тангенса (1.2):

$$\omega = 0, X_1 X_2 X_3, \quad (2.1)$$

где $X_i \in \{0-9\}$ будут стартовыми нетерминальными символами грамматик. Для каждого возможного значения X_i строится собственная порождающая грамматика G_i .

Например, для $X_i = 0$ — грамматика $G_0 = \langle T, N, P, 0 \rangle$, для $X_i = 1$ — грамматика $G_1 = \langle T, N, P, 1 \rangle$ и т.д.

Таким образом, из ω (2.1) мы получаем символьную цепочку типа: *aababbaaa*.

Шаг 3. Анализ цепочки и получение коэффициента симметричности.

Третьим шагом будет анализ данной цепочки на свойства переставной либо зеркальной симметрии [Александров, 1968] (сопоставление с символьными цепочками из памяти нейрона по определенному паттерну) и получение коэффициента симметричности k_s , который и будет являться выходом нейрона.

После прохождения 1-2 обучающих выборок значения, которые получены лишь один раз, будут удаляться из памяти, чтобы в дальнейшем туда могли попасть новые значения.

Обучение нейронной сети такого типа возможно методом обратного распространения ошибки. Однако в виду того, что мы не можем оптимизировать веса напрямую из выхода нейрона (коэффициента симметричности k_s), предлагается проводить оптимизацию весов по значению ω , которое мы использовали для генерации символьных цепочек.

Заключение

Предлагаемый подход представления функции активации как меры выполнения свойств симметрий между объектами пространства может сделать нейронные сети более доступными для понимания ввиду интерпретируемости сигналов. Появляется возможность сокращения числа скрытых слоев для получения качественного ответа, что может положительно повлиять на скорость работы сети [Ложкин и др., 2017b].

Следующим шагом будет проведение эксперимента по обучению нейронной сети с предлагаемой функцией активации на популярных тренировочных наборах данных MNIST, CIFAR и др.

В дальнейшем для развития данного подхода, стоит рассмотреть оставшиеся виды автоморфизмов и реализовать активацию на основе их свойств.

Список литературы

[Созыкин, 2017] Созыкин А.В. Обзор методов обучения глубоких нейронных сетей // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2017. Т. 6, № 3.

[Goodfellow et al., 2016] Goodfellow I., Bengio Y., Courville A. Deep Learning. The MIT Press, 2016

[He et al., 2016a] K. He, X. Zhang, S. Ren, J. Sun. Deep residual learning for image recognition. In Proceedings of CVPR, pp. 770–778, 2016

[Krizhevsky et al., 2012] A. Krizhevsky, I. Sutskever, G. Hinton. Imagenet classification with deep convolutional neural networks // Advances in Neural Information Processing Systems (NIPS), 2012.

[Ramachandran et al., 2017] Ramachandran, B. Zoph, Q. V. Le. Searching for activation functions. CoRR, 2017.

[Рудой, 2011] Рудой Г.И. Выбор функции активации при прогнозировании нейронными сетями // Машинное обучение и анализ данных. М., 2011. Т. 1, № 1.

[He et al., 2015b] K. He, X. Zhang, S. Ren, J. Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification // IEEE International Conference on Computer Vision (ICCV), 2015.

[Xu et al., 2015] B. Xu, N. Wang, T. Chen, M. Li. Empirical Evaluation of Rectified Activations in Convolution Network // ICML Deep Learning Workshop, 2015.

[Ложкин, 2010a] Ложкин А. Г. Симметрия как единое свойство пространства и живого организма // Тиетта. 2010. № 3(13).

[Bozek et al., 2018] Bozek P., Lozhkin A., Galajdova A., Arkhipov I., Maiorov K. Information technology and pragmatic analysis. Computing and informatics. 2018. Vol. 37, Issue 4.

[Волкова, 2009] И.А. Волкова, А.А. Вылиток, Т.В. Руденко. Формальные грамматики и языки. Элементы теории трансляции: Учебное пособие для студентов II курса (издание третье, переработанное и дополненное). — М.: Издательский отдел факультета ВМиК МГУ им. М.В.Ломоносова, 2009.

[Александров, 1968] Александров П.С. Лекции по аналитической геометрии, пополненные необходимыми сведениями из алгебры : С прил. собр. задач, снабжен. решениями, сост. А.С. Пархоменко. - М. : Наука, 1968.

[Ложкин и др., 2017b] Ложкин А.Г., Майоров К.Н. О некоторых проблемах разработки автономных роботов // Вестник ИжГТУ имени М. Т. Калашникова. 2017. №4.

УДК 004.932.7, 004.4, 004.021

**ВОССТАНОВЛЕНИЕ КАРТ ГЛУБИН
ИЗОБРАЖЕНИЙ В ЗАДАЧЕ ОДНОВРЕМЕННОГО
КАРТИРОВАНИЯ И ЛОКАЛИЗАЦИИ ПО
ВИДЕОПОТОКУ НА ВСТРАИВАЕМОЙ СИСТЕМЕ
NVIDIA JETSON TX2¹**

А.В. Боковой (*bokovoy@isa.ru*)

Федеральный исследовательский центр «Информатика и
управление» Российской академии наук, Москва

Российский университет дружбы народов, Москва

К.Ф. Муравьев (*muravev.kf@phystech.ru*)

Московский физико-технический институт, Москва

К.С. Яковлев (*yakovlev@isa.ru*)

Федеральный исследовательский центр «Информатика и
управление» Российской академии наук, Москва

Задача восстановления карт глубин изображений весьма актуальна в мобильной робототехнике, поскольку восстановление глубины может использоваться в контексте одновременного картирования и локализации по видеопотоку для получения метрической информации и построения карты с реальным масштабом. В работе представлен ряд улучшений, направленных на использование нейронных сетей для восстановления глубины в реальном времени на платформе NVidia Jetson TX2. Исследуются различные комбинации и реализации представленных улучшений для выбора решения, обеспечивающего приемлемое соотношение скорости и точности работы. Проводится эксперимент по построению карты неизвестного помещения в реальном времени на NVidia Jetson TX2.

Исходный код нейросетевых моделей, а также программный узел

¹Работа выполнена при финансовой поддержке РНФ (проект #16-11-00048 – разработка архитектур и проведение экспериментов) и при поддержке Программы РУДН «5-100» (дополнительная обработка результатов).

восстановления глубины для Robot Operating System доступен по ссылке: https://github.com/CnnDepth/tx2_fcnn_node

Ключевые слова: восстановление глубины, полносверточные нейронные сети, методы одновременного картирования и локализации по видеопотоку, NVidia Jetson TX2.

Введение

Восстановление глубины является одной из важных задач мобильной робототехники, дополненной реальности и других областей. В настоящее время существует большое количество датчиков, способных напрямую решать поставленную задачу, например сканирующие лидары, RGB-D камеры. Эти датчики, как правило, дорогие, габаритные, тяжелые и потребляют большое количество электроэнергии, что ограничивает их область применения, если речь идет о малых мобильных роботах. Такое ограничение привело к повышенному интересу к методам и алгоритмам восстановления глубины с использованием единственной камеры, поскольку камерами, как правило, оснащаются все робототехнические системы из-за их легкости, компактности и энергоэффективности.

В настоящее время существуют методы машинного обучения и архитектуры сверточных нейронных сетей, которые позволяют восстанавливать глубину изображения с приемлемой для задач мобильной робототехники точностью [Garg et al., 2016][Li et al., 2015]. Как правило, авторы этих методов сконцентрированы на точности восстановления, в то время как скорость работы выходит за рамки исследований. Поэтому, такие методы не могут быть использованы в первоначальном виде для создания полностью автономных робототехнических систем, которые не несут на борту мощных вычислителей, а имеют встраиваемый компьютер, пусть даже специально и предназначенный для задач обработки изображений, такой как NVidia Jetson TX2. Также, на текущий момент не существует полностью воспроизводимого (в контексте исходного кода) результата применения полносверточных нейронных сетей для одновременного картирования и локализации по видеопотоку (vision-based Simultaneous Localization and Mapping - vSLAM) в реальном времени на встраиваемых системах. Описанные выше проблемы определяют тему представленной статьи.

Основной целью работы является создания метода восстановления глубины, основанного на нейросетевом подходе, который: а) обеспечивает точность работы, достаточную для использования в методах одновременного картирования и локализации, б) способен функционировать в режиме реального времени на встраиваемой

платформе NVidia Jetson TX2, в) открыт сообществу, т.е. имеет открытый исходный код в виде узла фреймворка Robot Operating System (ROS).

1. Постановка задачи

1.1. Восстановление глубины единственного изображения

Проблема восстановления глубины изображений изучена достаточно хорошо. До того, как нейронные сети стали популярными, для решения задачи использовались методы предобработки изображений, выделения особенностей, выделения границ и т.д. К примеру, в работе [Saxena et al., 2008] используются Марковские цепи (Markov Random Field - MRF) для получения информации об отношении расстояний разных областей изображений друг к другу. К разным областям исходного изображения применяются заданные вручную фильтры, затем полученная информация используется MRF для нахождения отношений между обработанными областями.

На текущий момент, для восстановления глубины широко используются полносверточные нейронные сети (Fully-convlolutional neural networks - FCNN) [Long et al., 2015]. Типичная архитектура FCNN состоит из энкодера и декодера. Энкодер, как правило является обычной сверточной нейронной сетью, которая используется для задачи классификации изображений [Солдатова и др., 2010], но без полносвязного слоя. Энкодер используется для извлечения набора особенностей изображений, которые затем преобразуются декодером в новое изображение (карту глубины) последовательным использованием операций повышения дискретизации (up-convolution), развертки (deconvolution) и др.

В работе [Laina et al., 2016] предложен новый алгоритм повышения дискретизации в составе декодера, а также использована обратная функция потерь Huber [Owen, 2007]. Предложенная в статье архитектура протестирована для задачи восстановления глубины изображений в реальном времени, но целевой платформой является графический ускоритель NVidia GeForce GTX TITAN с 12 GB памяти, который гораздо мощнее, чем графический ускоритель на платформе NVidia Jetson TX2. Анализ этой и других работ показывает, что в настоящее время существует множество работ, связанных с восстановлением глубины изображений с помощью FCNN, но целевыми системами являются мощные рабочие станции, а не встраиваемые системы.

1.2. Восстановление глубины на встраиваемых системах

В последнее время обретают популярность встраиваемые системы, которые с проектированы для задач, решаемых нейросетевыми методами. Одной из таких платформ является NVidia Jetson TX2, которая используется для ускорения обработки изображений на графическом процессоре в задачах семантической сегментации [Pannu et al., 2015], распознавания объектов, классификация изображений и т.д. Размер и энергопотребление такой платформы позволяет использовать ее в качестве бортового вычислителя робототехнической системы. Доступность графического ускорителя стало решающим фактором при выборе NVidia Jetson TX2 в качестве целевой платформы, используемой в представленной работе для исследований FCNN.

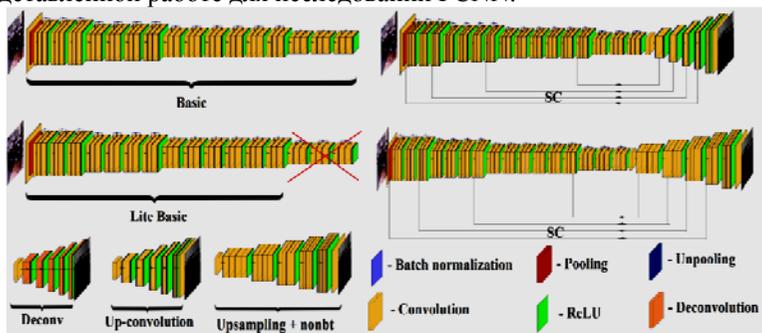


Рис.1 Визуализация исследованных архитектур.

В настоящий момент, уже существует исследование на тему использование FCNN для восстановление глубины изображений на платформе NVidia Jetson TX2 в реальном времени [Spek et al., 2018]. Авторы используют облегченную архитектуру энкодер-декодер, которая была обучена с помощью переноса знаний с более глубокой архитектуры. Было достигнуто быстроедействие в 30 кадров в секунду с сравнимой с актуальными методами восстановления глубины точностью. Но исследования применимости в задаче vSLAM не проводились (только устранение дрейфа масштаба карты), а результаты работы – не воспроизводимы (в контексте исходного кода или бинарных фалов). Исследование, представленное же в данной статье нацелено на применение нейронных сетей для восстановления глубины изображений в реальном времени на встраиваемой системе. Главной задачей является применение в методах vSLAM и воспроизводимость полученных результатов.

2. Исследованные архитектуры

В работе были исследованы разные модификации архитектур полносверточных нейронных сетей для задачи восстановления глубины изображений (см. рис.1). Поскольку основной задачей была работа в реальном времени, был реализован ряд улучшений, чтобы повысить быстродействие и сохранить точность работы алгоритмов.

2.1. Описание архитектур

В качестве основного энкодера используется ResNet50 (и его облегченная версия) и несколько разных модификаций декодера. Также, были предложены некоторые улучшения для увеличения быстродействия блоков энкодера. Таким образом, были получены 6 целевых архитектур для исследований.

1) **Энкодер.** ResNet50 является одним из самых эффективных энкодеров для выделения набора особенностей изображений. Несмотря на то, что архитектура является достаточно глубокой (50 слоев) с несколькими остаточными (residual) блоками, ее быстродействия хватает для работы в реальном времени. Выходом стандартной архитектуры ResNet для изображения размером $640 \times 480 \times 3$ является набор особенностей размера $20 \times 15 \times 2048$. Обозначим эту архитектуру как **Basic**.

Также, была исследована облегченная версия архитектуры ResNet50 без последнего набора остаточных слоев. Выход такой сети имеет размер $30 \times 40 \times 1024$. Такая архитектура работает гораздо быстрее исходной, при этом потери в точности – не значительные. Облегченная версия архитектуры обозначена как **Lite Basic**.

2) **Декодер.** Для повышения дискретизации, были исследованы алгоритмы, позволяющие ускорить обработку изображений при сохранении точности на подходящем для задачи vSLAM уровне. Первая модификация декодера состоит из разверточных (**Deconv**) слоев для преобразования набора особенностей изображения в карту глубины.

Вторая модификация включает использование слоя повышения дискретизации с последующей серией одномерных свертков (non-bottleneck ID convolution), которая использовалась ранее в сети ERFNet. Повышение дискретизации происходит алгоритмом поиска ближайшего соседа, затем идет 2 последовательных набора свертков размера 3×1 и 1×3 , затем нормализация (batch normalization). Выход активируется функцией ReLU. Такая модификация обозначена как **Upsampling + nonbt**.

Третья модификация – использование слоев **up-convolution** в декодере. Такой декодер состоит из серии операций повышения

дискретизации и свертки. Также, было исследована применимость операции interleaving (**interl**) для увеличения быстродействия. Такой подход позволяет в случае необходимости перенести знания с более ресурсоемкой архитектурой *up-convolution* (**T**).

3) **Проекции (shortcuts)**. В работе также были использованы проекции (обозначены как **SC**) из слоев энкодера в декодер. Такой подход позволяет повысить точность восстановления глубины на границах объектов.

2.2. Функция потерь

В процессе обучения нейронных сетей, было установлено, что ошибка на близких расстояниях больше чем средняя ошибка по всей предсказанной карте глубины. Для того, чтобы уменьшить такую ошибку, может быть использована функция потерь Huber (и ее модификация VerHu). Но пороговое значение, которое разграничивает «близкие» и «далекие» расстояния в функциях Huber и VerHu не позволяют эффективно решить эту проблему.

Таблица 1. Результаты тестирования архитектур восстановления глубины изображений

Name	Loss	Input resolution	Decoder	MSE	REL	δ^1	δ^2	δ^3	PC time (s)	Jetson time (s)
Basic	BerHu	640x480	Deconv	0.467	0.186	0.718	0.929	0.980	0.144	0.152
Basic + SC	BerHu	640x480	Deconv	0.487	0.194	0.695	0.915	0.975	0.521	0.563
Basic + SC	aBerHu	640x480	Upsampling + nonbt	0.440	0.184	0.725	0.932	0.982	0.158	0.215
Basic + SC	MSE + REL	640x480	Upsampling + nonbt	0.419	0.173	0.748	0.944	0.987	0.158	0.215
Basic + SC	MSE + REL	320x240	Upsampling + nonbt	0.408	0.180	0.746	0.940	0.984	0.049	0.062
Lite Basic + SC	MSE + REL	320x240	Upsampling + nonbt	0.533	0.202	0.687	0.915	0.979	0.035	0.049
Basic + SC + interl	MSE + REL	640x480	Up-convolution	0.514	0.206	0.708	0.912	0.970	0.285	0.328
Basic + SC + interl	MSE + REL	320x240	Up-convolution	0.580	0.215	0.673	0.899	0.965	0.057	0.067
Basic + SC + interl + T	MSE + REL	640x480	Up-convolution	0.445	0.178	0.714	0.939	0.987	0.181	0.227
Basic + SC + interl + T	MSE + REL	320x240	Up-convolution	0.495	0.181	0.724	0.940	0.983	0.048	0.061
Lite Basic + interl + T	MSE + REL	640x480	Up-convolution	0.658	0.233	0.642	0.886	0.964	0.101	0.135
Lite Basic + interl + T	MSE + REL	320x240	Up-convolution	0.660	0.236	0.649	0.881	0.960	0.027	0.037

Таким образом, был предложен адаптивный (**aBerHu**) подход для вычисления порогового значения (k) в таких функциях. На каждом шаге обучения происходит сравнение предсказанных глубин, лежащих в отрезках $[k - d; k]$ и $[k + d; k]$. Затем, граница передвигается в сторону наибольшего значения $k \pm lr * d$. Наряду с функцией aBerHu используется функция **MSE+REL**.

3. Результаты экспериментов

Разработанные архитектуры были исследованы на встраиваемой платформе NVidia Jetson TX2. Архитектуры были реализованы с помощью фреймворков Tensorflow и Keras на языке Python. Операции

interleaving, а также up-convolution были реализованы для CPU и GPU на языке C/C++ с помощью компиляторов g++ и nvcc соответственно. Для обучения был использован гибридный высокопроизводительный вычислительный комплекс ФИЦ ИУ РАН [ФИЦ ИУ РАН, 2018].

3.1. Коллекция данных

На этапе обучения и валидации была использована открытая коллекция данных NYU Dataset v2, которая состоит из более чем 400 000 пар изображение-глубина, заснятых на более 470 разных сценах. Для увеличения разнообразия изображений из NYU Dataset v2 были случайно обрезаны (вместо с соответствующими ими картами глубин), а также случайно отображены и повернуты.

3.2. Исследование архитектур

Результаты тестирования описанных в статье архитектур представлено в таблице 1. Лучший результат в плане быстродействия на NVidia Jetson TX2 показала архитектура **Lite Basic + interl + T** (37 мс). Время обработки сравнимо с методом, представленным в [Long et al., 2015], при этом отмечается незначительная для задачи vSLAM потеря в качестве.

С другой стороны, в архитектурах **Basic + SC + Upsampling + nonbt** удалось добиться лучшего показателя REL, чем в некоторых архитектурах из [Laina et al., 2016][Liu et al., 2016], которые не нацелены на работу в реальном времени. При этом, один кадр обрабатывается 62мс, что достаточно для использования в задаче vSLAM.

Архитектура **Lite Basic** дает большой прирост к скорости работы, но при этом теряет в точности, даже с переносом знаний с более тяжелой архитектуры. Потеря в точности может быть компенсирована модификациями, представленными в статье (**Upsampling + nonbt, Up-convolution**) и ускорены с помощью операции **interl**.

3.3. Применение в задаче vSLAM

Для исследования применимости разработанных архитектур в задаче vSLAM, был реализован узел для фреймворка Robot Operating System (ROS). Для картирования и локализации был использован открытый узел RTAB-Map. Все вычисления выполняются автономно на NVidia Jetson TX2. Эксперименты показали, что разработанные архитектуры позволяют точно и быстро производить восстановление карты и вычисление позиций системы на встраиваемой платформе. FCNN работает на графическом процессоре со скоростью 16 кадров в секунду, в то время как алгоритм vSLAM – на центральном, поэтому алгоритмы не занимают ресурсы друг у друга.

ROS позволяет выполнять картирование и локализацию в двух сценариях: 1) нейронная сеть работает на NVidia Jetson TX2, затем результаты отсылаются на рабочую станцию и vSLAM выполняется уже там, 2) оба алгоритма работают на NVidia Jetson TX2. Оба сценария подразумевают выполнение в реальном времени. Код всех исследований, а также реализация для TensorRT и ROS доступны по ссылке: <https://github.com/cnndepth>. Видео с демонстрацией результата доступно по ссылке: <https://www.youtube.com/watch?v=ayjvfm-C7s>.

Заключение

В работе представлены результаты исследования разных модификаций архитектур для восстановления глубины на NVidia Jetson TX2. Исследование показало, что предложенные архитектуры могут работать в реальном времени при сравнимой с современными архитектурами точностью. Также, был разработан узел для фреймворка ROS и протестирован для картирования и локализации в помещении с помощью узла RTAB-Map. Тестирование показало, что архитектуры могут быть использованы автономно на встраиваемой системе для картирования и локализации по видеопотоку.

Список литературы

[Андрейчук и др., 2016] Андрейчук А. А., Боковой А. В., Яковлев К. С. Оценка быстродействия некоторых алгоритмов планирования траектории на широко используемой в робототехнике платформе RaspberryPi // Экстремальная робототехника. – 2016. – Т. 1. – №. 1. – С. 184-189.

[Солдатова и др., 2010] Солдатова О. П., Гаршин А. А. Применение сверточной нейронной сети для распознавания рукописных цифр //Компьютерная оптика. – 2010. – Т. 34. – №. 2.

[ФИЦ ИУ РАН, 2018] Федеральный исследовательский центр Информатика и управление РАН [Электронный ресурс]: сайт. – Москва: ФИЦ ИУ РАН. – URL: <http://hhpcc.frccsc.ru> (дата обращения: 12.09.2018)

[Garg et al., 2016] Garg, R., BG, V. K., Carneiro, G., & Reid, I. Unsupervised cnn for single view depth estimation: Geometry to the rescue //European Conference on Computer Vision. – Springer, Cham, 2016. – С. 740-756.

[Laina I. et al., 2016] Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., & Navab, N. Deeper depth prediction with fully convolutional residual networks //2016 Fourth international conference on 3D vision (3DV). – IEEE, 2016. – С. 239-248.

[Li et al., 2015] Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – С. 1119-1127.

[Liu et al., 2016] Liu, F., Shen, C., Lin, G., & Reid, I. Learning depth from single monocular images using deep convolutional neural fields //IEEE transactions on pattern

analysis and machine intelligence. – 2016. – T. 38. – №. 10. – C. 2024-2039.

[Long et al., 2015] Long J., Shelhamer E., Darrell T. Fully convolutional networks for semantic segmentation //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – C. 3431-3440.

[Owen, 2007] Owen A. B. A robust hybrid of lasso and ridge regression //Contemporary Mathematics. – 2007. – T. 443. – №. 7. – C. 59-72.

[Pannu et al., 2015] Pannu G. S., Ansari M. D., Gupta P. Design and implementation of autonomous car using Raspberry Pi //International Journal of Computer Applications. – 2015. – T. 113. – №. 9.

[Paszke et al., 2016] Paszke, A., Chaurasia, A., Kim, S., & Culurciello, E. Enet: A deep neural network architecture for real-time semantic segmentation // arXiv preprint arXiv:1606.02147. – 2016.

[Saxena et al., 2008] Saxena A., Chung S. H., Ng A. Y. 3-d depth reconstruction from a single still image //International journal of computer vision. – 2008. – T. 76. – №. 1. – C. 53-69.

[Spek et al., 2018] Spek A., Dharmasiri T., Drummond T. CReaM: Condensed Real-time Models for Depth Prediction using Convolutional Neural Networks // 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). – IEEE, 2018. – C. 540-547.

РАЗРАБОТКА МОДУЛЬНОЙ ПРЕЦЕДЕНТНОЙ СИСТЕМЫ ДЛЯ РЕШЕНИЯ ЗАДАЧ КЛАССИФИКАЦИИ ДАННЫХ¹

П.Р. Варшавский (*varp@appmat.ru*)

Р.В. Алехин (*r.alekhin@gmail.com*)

Т.О. Блашонков (*timolb@yandex.ru*)

И.Е. Мухачева (*muirishka@yandex.ru*)

С.А. Поляков (*serchos96@yandex.ru*)

ФГБОУ ВО «Национальный исследовательский университет
«МЭИ», Москва

В работе рассматриваются вопросы, связанные с разработкой архитектуры модульной системы поиска решения на основе прецедентов, способной выполнять все этапы цикла обучения на основе прецедентов и её реализацией в виде веб-приложения. Также подробно рассмотрены основные модули системы и показан пример её использования для решения задачи классификации данных.

Ключевые слова: машинное обучение, анализ данных, классификация данных, прецедентный подход, алгоритм k -NN

Введение

В настоящее время возрастает потребность в анализе больших данных для различных прикладных областей, таких как медицина, распознавание образов и речи, биоинформатика, бизнес и т.д. Одним из перспективных и актуальных методов анализа больших данных на сегодняшний день является применение машинного обучения – класса методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи, а обучение в процессе применения решений множества сходных задач. Одним из подходов, широко используемых в задачах машинного обучения, является подход на основе прецедентов (Case-Based Reasoning – CBR).

В данной работе рассматриваются различные вопросы, связанные с

Работа выполнена при финансовой поддержке РФФИ (проект № 18-07-00050, № 18-29-22019)

разработкой программных средств, реализующих модульную прецедентную систему (CBR-систему), предназначенную для исследования основных методов машинного обучения на основе прецедентов и решения задач классификации, кластеризации, прогнозирования и др.

1. Рассуждения на основе прецедентов

Прецедент (от латинского *praecedentis* – предшествующий) определяется как случай, имевший место ранее и служащий примером или оправданием для последующих случаев подобного рода [Еремеев, Варшавский, 2008].

Рассуждения на основе прецедентов (CBR) является подходом, позволяющим решить новую, неизвестную задачу, используя или адаптируя решение уже известной задачи, т.е. используя уже накопленный опыт решения подобных задач.

Для эффективного применения CBR-систем не требуется глубокого анализа предметной области, достаточно указать проблему и ее решение путем предоставления нескольких примеров, аналогичных случаев и ссылок на некоторое сходство.

Процесс вывода на основе прецедентов включает четыре основных этапа, образующих цикл рассуждения на основе прецедентов или CBR-цикл.

Основными этапами CBR-цикла являются:

- *извлечение* наиболее похожего прецедента (или прецедентов) для сложившейся ситуации из библиотеки прецедентов (БП);
- *повторное использование* извлеченного прецедента для попытки решения текущей проблемы;
- *пересмотр и адаптация* в случае необходимости полученного решения в соответствии с текущей проблемой;
- *сохранение* вновь принятого решения как части нового прецедента.

2. Способы представления и извлечения прецедентов

В общем случае модель представления прецедента включает описание ситуации, решение для данной ситуации и результат применения [Варшавский и др., 2015]:

$CASE = (Situation, Solution, Result)$,

где *Situation* – ситуация, описывающая данный прецедент, а *Solution* – решение (например, диагноз и рекомендации пользователю), *Result* – результат применения решения.

Основные способы представления прецедентов можно разделить на следующие группы: параметрические; объектно-ориентированные; специальные (в виде деревьев, графов, логических формул и т.д.).

Для программной реализации прецедентной системы в качестве базовой модели представления прецедентов была выбрана параметрическая модель. Выбор параметрической модели обусловлен простотой ее реализации и меньшими затратами на поддержание и сопровождение параметрической БП, в отличие от БП на основе других методов представления прецедентов.

Для успешной реализации рассуждений на основе прецедентов необходимо обеспечить корректное извлечение прецедентов из БП [Рассел, Норвиг, 2005]. Выбор метода извлечения прецедентов напрямую связан со способом представления прецедентов и соответственно со способом организации БП.

Существует целый ряд методов извлечения прецедентов и их модификаций: метод k ближайших соседей, метод извлечения на основе деревьев решений, метод извлечения с учётом применимости прецедентов, метод извлечения с использованием аппарата искусственных нейронных сетей и др.

Каждый из этих методов обладает своими преимуществами и недостатками. И поэтому выбор метода решения зависит от специфики конкретной задачи и располагаемых ресурсов, например, времени, объемов памяти вычислительного кластера и т.д.

В качестве алгоритма извлечения прецедентов из БП был выбран и реализован алгоритм k ближайших соседей (k - NN) [S. V. Imandoust, M. Bolandraftar, 2013].

Преимущества алгоритма k - NN для извлечения прецедентов:

- нужно знать только функцию близости между объектами;
- может быть применен к объектам любой структуры, если задана функция близости;
- простая логика работы и реализация;
- разумное объяснение результатов классификации.

Недостатки алгоритма k - NN :

- с ростом БП снижается быстродействие и увеличиваются требования по памяти;
- точность ухудшается с ростом размерности пространства;
- проблема выбора метрики.

Для определения ближайшего прецедента были использованы следующие метрики.

Евклидова метрика (евклидово расстояние): $d_{CT} = \sqrt{\sum_{i=1}^n (x_{iC} - x_{iT})^2}$.

L1 (Манхэттенская расстояние): $d_{CT} = \sum_{i=1}^n |x_{iC} - x_{iT}|$.

L^∞ (Расстояние Чебышева): $d_{CT} = \max_i |x_{iC} - x_{iT}|$.

Расстояние Минковского: $d_{CT} = (\sum_{i=1}^n |x_{iC} - x_{iT}|^p)^{\frac{1}{p}}$.

Также в системе был реализован автоподбор метрики при наличии тестовой выборки данных. Система последовательно выполняет алгоритм k -NN, используя приведенные выше метрики, и затем выбирает ту, для которой алгоритм дает наибольшую точность на тестовом наборе данных.

3. Способы повышения эффективности работы СBR-систем

СBR-система может содержать избыточное число прецедентов, что ведет к снижению эффективности работы системы, поэтому существуют ряд методов, направленных на повышение эффективности работы прецедентных систем [Варшавский и др., 2017]:

- индексирование БП;
- сокращение количества прецедентов в БП;
- с использованием методов классификации;
- на основе кластерных методов;
- обобщение накопленного опыта (прецедентов) с использованием индуктивных методов;
- использование аппарата искусственных нейронных сетей и др.

Одним из способов сокращения количества прецедентов в БП является классификация с последующим удалением всех прецедентов, которые определяются как ближайšie к выбранному прецеденту.

Это позволяет представить БП в компактном виде для пользователя с сохранением распределения по классам и существенно повысить быстродействие СBR-систем, но при этом возможно снижение качества получаемого решения.

Процесс такого сокращения состоит из следующих основных этапов:

1. начиная с первого прецедента, из БП последовательно выбираются прецеденты и для них выполняются следующие этапы. Процесс завершается, когда все прецеденты из БП будут рассмотрены и удалены ближайšie к ним;

2. определяется множество ближайших прецедентов к выбранному на предыдущем этапе с заранее заданной пользователем степенью сходимости;

3. найденное множество ближайших прецедентов удаляется из БП и возврат к первому этапу.

Другой способ сокращения количества прецедентов в БП базируется на методах кластеризации. Кластеризация – задача разбиения заданной выборки объектов (ситуаций) на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из схожих объектов, а объекты разных кластеров существенно отличались. При этом количество кластеров может быть не задано и определяться в процессе работы алгоритма.

Сокращение количества прецедентов в БП на основе кластеризации включает следующие этапы.

1. В качестве начальных центров масс кластеров выбираются случайные прецеденты, число которых может соответствовать количеству заранее известных классов (кластеров).

2. Для каждого центра масс кластера определяется множество прецедентов, для которых расстояние от прецедента до центра кластера минимально.

3. В каждом полученном множестве ближайших прецедентов происходит пересчет центров масс кластера.

4. Вычисляется суммарное расстояние от прецедентов до полученных центров. Если суммарное расстояние не уменьшилось, тогда переход к пункту 5, иначе возврат к пункту 2.

5. На последнем этапе из БП удаляются все прецеденты за исключением полученных центров кластеров.

4. Модульная CBR-система

Архитектура системы

Архитектура разработанной модульной CBR-системы состоит из следующих основных компонентов (рис. 1) [Варшавский и др., 2015]:

- пользовательский интерфейс – интерфейс для взаимодействия с экспертом или пользователем и отображения результатов работы;
- модуль извлечения прецедентов – в системе реализован ряд методов для извлечения прецедентов из БП (алгоритм NN, k-NN);
- модуль представления прецедентов (знаний) – содержит ряд методов для создания, редактирования и просмотра БП, а также информации о структуре прецедентов;
- модуль оптимизации БП – предназначен для сокращения количества прецедентов в БП с использованием различных классификационных (на основе алгоритма k-NN) и кластерных алгоритмов (на основе алгоритма k-средних) [Алехин и др., 2018].

Разработанная система представляет собой веб-приложение,

реализованное на языке python 3.4. с использованием веб-фреймворка flask. Flask — фреймворк для создания веб-приложений на языке программирования Python. Относится к категории так называемых микрофреймворков — минималистичных каркасов веб-приложений, сознательно предоставляющих лишь самые базовые возможности [Гринберг, 2014]. Для хранения прецедентов и администрирования пользователей используется СУБД MySQL. Для разработки пользовательского интерфейса был использован язык разметки HTML, каскадные таблицы стилей CSS и язык программирования JavaScript, а также его библиотеки JQuery, Ajax и фреймворк Bootstrap 3.

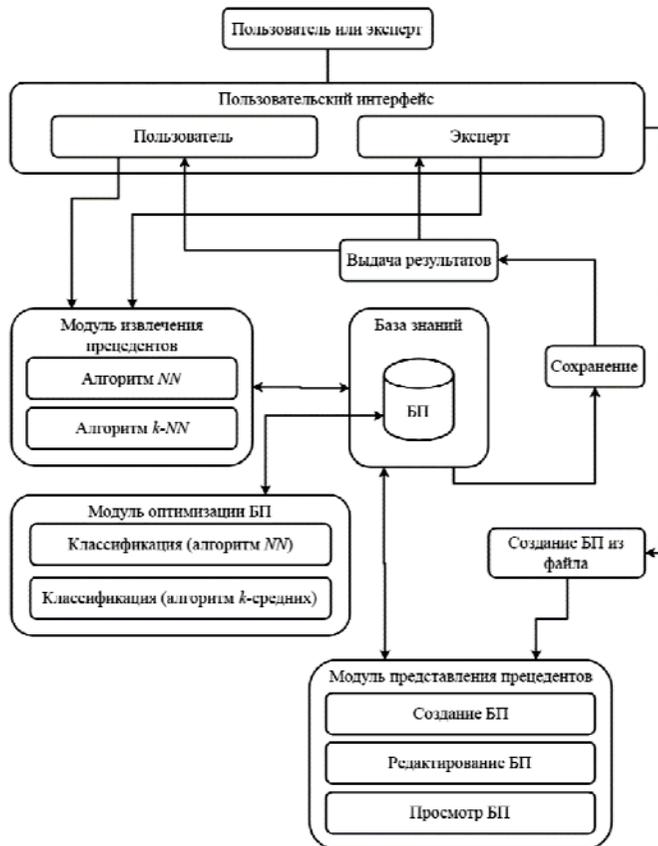


Рис. 1. Архитектура разработанной модульной CBR-системы

Разработанная модульная CBR-система позволяет выполнять редактирование и анализ данных, сравнивать различные методы их обработки, с целью определения наилучшего для решения поставленной задачи.

Для работы приложения необходимо наличие БП. Для этого нужно создать структуру будущей БП и внести в нее прецеденты, а также можно загрузить данные из файла формата Comma-Separated Values (.csv) или Text (.txt), содержащего набор данных. В этом случае программа автоматически сгенерирует структуру БП и заполнит ее данными из файла. В случае необходимости предусмотрена возможность редактирования БП с помощью изменения существующих прецедентов, удаления и добавления новых. Можно изменить структуру БП, путем удаления существующих или добавления новых полей.

Апробация системы

Рассмотрим работу CBR-системы на примере решения задачи классификации данных. Для этого воспользуемся набором данных «Wine Quality», взятым из открытого репозитория «UCI Machine Learning Repository».

БП системы была сформирована на основе набора данных, состоящего из 4898 проб белого вина, каждый из которых имеет 12 атрибутов, последний из них – качество рассматриваемого образца (от 0 до 10), а также из репозитория в CBR-систему был загружен тестовый набор данных, состоящий из 210 проб вина. Рассмотрим подробнее атрибуты: fixed acidity – фиксированная кислотность; volatile acidity – летучая кислотность; citric acid – лимонная кислота; residual sugar – остаточный сахар; chlorides – хлориды; free sulfur dioxide – свободный диоксид серы; total sulfur dioxide – общий диоксид серы; density – плотность; pH – водородный показатель; sulphates – сульфаты; alcohol – спирт.

Выходная переменная (на основе сенсорных данных): quality – качество (оценка от 0 до 10).

В результате загрузки данных из репозитория в CBR-систему была сформирована БП, которая при решении задачи классификации данных на тестовом наборе с использованием алгоритма k-NN (при k равном 7) и Евклидовой метрики, выбранной автоматически, показала точность классификации 62% (рис. 2). Затем, используя модуль оптимизации БП, удалось сократить число прецедентов до 2579 и получить точность классификации равную 59% с использованием алгоритма классификации k-NN и сократить число прецедентов до 11 и получить точность

классификации равную 60% с использованием алгоритма кластеризации k-средних.

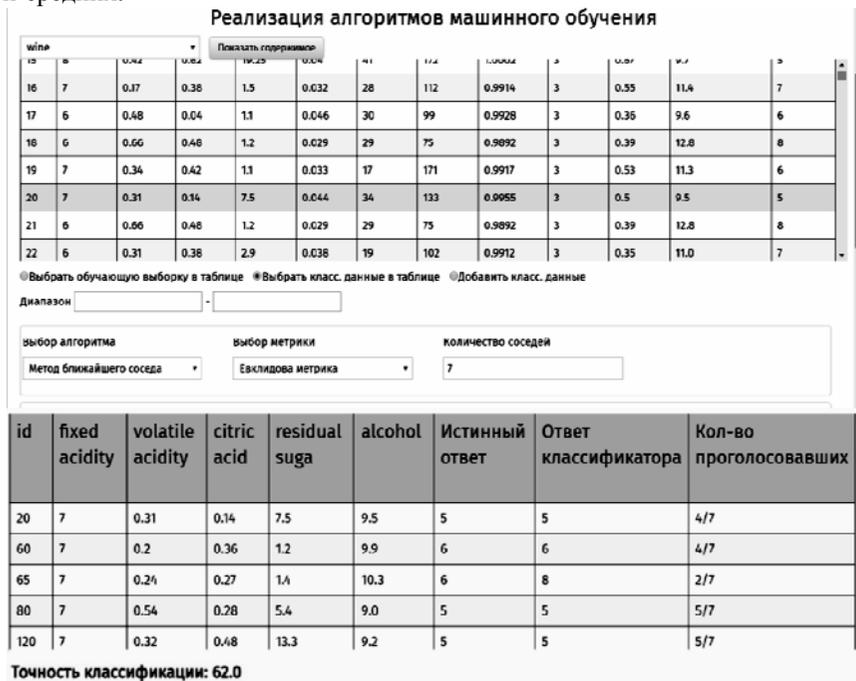


Рис. 2. Результат классификации по методу k-NN

Заключение

В настоящее время все чаще встречаются ситуации, когда поставленную задачу крайне сложно решить из-за большого объема данных и сложных взаимосвязей. По этой причине приходится прибегать к использованию методов и средств машинного обучения и, в частности, методам на основе прецедентов. Указанные средства и технологии помогают в решении задач подобного рода. В работе основное внимание уделено описанию возможностей и особенностей рассуждений на основе прецедентов и реализованному приложению (СБР-системе), применяемому для решения задач классификации данных. Работа реализованной системы была рассмотрена на примере решения задачи классификации данных из набора «Wine Quality», взятом из открытого репозитория «UCI Machine Learning Repository».

Список литературы

[**Варшавский, Еремеев, 2008**] Варшавский П.Р., Еремеев А.П. Моделирование рассуждений на основе прецедентов в интеллектуальных системах поддержки принятия решений // Искусственный интеллект и принятие решений. 2009. № 2. С. 45-47.

[**Варшавский и др., 2015**] Варшавский П.Р., Алехин Р.В., Ар Кар Мью, Зо Лин Кхаинг Реализация прецедентного модуля для интеллектуальных систем // Программные продукты и системы / Software & Systems. – 2015. - № 2. – С. 26-31.

[**S. B. Imandoust, M. Bolandraftar, 2013**] Application of K-Nearest Neighbor (KNN) Approach for Predicting Economic Events: Theoretical Background // S B Imandoust et al. Int. Journal of Engineering Research and Applications Vol. 3, Issue 5, Sep-Oct 2013, pp.605-610

[**Варшавский и др., 2017**] Варшавский П.Р., Ар Кар Мью, Шункевич Д.В. Применение методов классификации и кластеризации для повышения эффективности работы прецедентных систем // Программные продукты и системы, No4, 2017, С. 625-631.

[**Рассел, Норвиг, 2007**] Рассел С., Норвиг П. Искусственный интеллект. Современный подход. — 2 изд.. М.: Издательство «Вильямс», 2007. 1410 с.

[**Алехин и др., 2018**] Алехин Р.В., Блашонков Т.О., Мухачева И.Е., Поляков С.А. Реализация модульной СВР-системы для изучения студентами принципов машинного обучения на основе прецедентов // Труды Международной научно-методической конференции. Информатизация инженерного образования. 2018. С. 1-4.

[**Гринберг 2014**] Гринберг М. Разработка веб-приложений с использованием Flask на языке Python // пер. с англ. А. Н. Киселева – М: ДМК Пресс, 2014. – 272 с.

УДК 004.8

КАРТИНА МИРА АГЕНТА-ОНКОЛОГА ДЛЯ РЕШЕНИЯ ЗАДАЧИ ДИАГНОСТИКИ РАКА ПОДЖЕЛУДОЧНОЙ ЖЕЛЕЗЫ¹

А.И. Молодченков (*aim@tesyan.ru*)

Федеральный исследовательский центр «Информатика и
управление» РАН, Москва

Российский университет дружбы народов, Москва

М.О. Зубрихина (*m.zubrikhina2014@yandex.ru*)

Российский университет дружбы народов, Москва

В работе описывается знаковая картина мира агента-онколога для решения задачи моделирования работы консилиума врачей при диагностике рака поджелудочной железы. Консилиум врачей состоит из нескольких специалистов и одного лидера, каждый из которых является автономным агентом со своей картиной мира. Результаты статьи лягут в основу построения алгоритма принятия решения коллективом этих агентов.

Ключевые слова: коллектив агентов, знак, знаковая картина мира, диагностика, консилиум врачей, искусственный интеллект, когнитивные исследования, рак поджелудочной железы

Введение

Анализ развития технологий моделирования поведения коалиций агентов и развития многоагентных систем показывает, что в этой области все больше внимания уделяется автономности самих агентов. Для этого применяются различные методы искусственного интеллекта, машинного обучения, когнитивных исследований и разрабатываются соответствующие архитектуры автономных агентов [Panov A. I., et c. 2017, Kim J. H. et al 2015, Albus J. S., 2002, Yoo J. K., et al 1984].

Архитектуры автономных агентов используются в робототехнике при решении таких задач, как планирование траектории, перемещение

¹ Работа выполнена при финансовой поддержке РФФИ (проект № 17-29- 07031).

робота из одной точки в другую, обнаружение пожаров и т.п. Однако в последние годы появляются исследования и разработки, которые позволяют интеллектуальным агентам решать более широкий класс задач, начиная от оператора, отвечающего на вопросы по телефону, или чат бота и заканчивая моделированием поведения коалиций агентов при решении различного рода научных и бизнес задач [Rosenfeld A. et al, 2017, Chae S. W, 2015].

В медицине тоже есть ряд задач, при решении которых могут быть использованы коалиции агентов. Одной из таких задач является моделирование работы консилиума врачей. В этом случае каждый врач консилиума рассматривается как отдельный агент. Проводятся исследования по решению этой задачи. Однако пока были построены только правила поведения консилиума в целом, при этом агенты консилиума не являются автономными.

В отличие от многих других задач, в консилиуме обязательно присутствует лидер, который принимает окончательное решение. Лидер и остальные врачи подбираются в зависимости от решаемой задачи. В этой работе описана картина мира агента-онколога при моделировании работы консилиума врачей для решения задачи диагностики рака поджелудочной железы.

1. Картина мира агента-онколога

В данной главе описана знаковая картина мира агента-онколога для решения задачи моделирования деятельности врачебного консилиума при диагностике рака поджелудочной железы. В основе знаковой картины мира, описанной в работах [Осипов Г. С. и др, 2017, Осипов Г. С. и др, 2014], лежит понятие знака, компоненты которого на синтаксическом уровне определяются следующим образом.

Пусть:

А - множество смыслов (как личностных, так и биологических),

М - множество значений,

Р - множество признаков объектов

Тогда:

$a \in A$ — подмножество множества личностных смыслов (возможно пустое),

$m \in M$ — подмножество множества значений (функциональное либо культурно—историческое),

$p \in P$ — подмножество множества признаков (перцепт либо образ)

агента-онколога формируется образ знака. В нашем случае в качестве знаков будут выступать медицинские термины и понятия, используемые

агентом-онкологом для решения поставленной задачи. Значениями знака являются функциональные назначения предмета, т.е. способы его использования и действия самого знака (например, значением знака «рак поджелудочной железы» выступает «метастазирует»). В качестве значений знака «Поджелудочная железа» могут выступать такие понятия, как «выработка ферментов, участвующих в пищеварении», «выработка гормонов» и т.д. Следует также отметить, что значения выступают декларативным знанием, то есть являются некими установленными в обществе фактами. Из этого следует, что в качестве значений можно взять понятия, описанные в медицинских справочниках, статьях и учебниках.

Подмножество личностных смыслов A , возникает благодаря опыту действия с предметом. Они могут быть представлены, например, тем опытом, который имеет врач в использовании некоторого способа диагностики. Получить данные о его опыте можно путем его интервьюирования, этот подход описан в книге [Осипов Г. С, 2018], и извлечения информации из историй болезни.

На множестве каждой компоненты знака генерируется своя семантическая сеть с заданным на ней семейством отношений:

H_p – семантическая сеть на множестве образов,

H_a – семантическая сеть на множестве смыслов,

H_m – семантическая сеть на множестве значений

Семиотической сетью называется тройка объектов $\Omega = (H_p, H_a, H_m)$, на которой заданы следующие операторы связывания:

$\Psi^m: 2^p \rightarrow 2^m$ – оператор связывания образов p со значениями m .

$\Psi^a: 2^m \rightarrow 2^a$ – оператор связывания значений со смыслами.

$\Psi^p: 2^a \rightarrow 2^p$ – оператор связывания смыслов с образами. [Осипов Г. С. и др, 2014]

На множестве образов заданы такие отношения, как отношения эквивалентности, включения, сходства и противопоставления, отношение «являться элементом образа» а также операция обобщения. На множестве значений заданы также отношения эквивалентности, включения, сходства и противопоставления, отношение «являться элементом значения», классификации, сценарное отношение и операция замыкания по значению. К семейству отношений на множестве личностных смыслов относятся отношения «являться элементом смысла», эквивалентности, сходства, включения, противопоставления, ситуационному отношению и операция агглютинации смыслов.

Приведем примеры отношений на множестве образов, значений и несколько основных знаков, которые будут представлены в знаковой картине агента-онколога. В качестве знаков выступают «Поджелудочная

железа», «Рак поджелудочной железы», «Компьютерная томография».

Для описания образной компоненты «Рак поджелудочной железы» выступают критерии, описанные в таблицах 1 и 2. Для каждого критерия из Табл. 1 определен вид семантической связи для дальнейшего использования в семантической сети на множестве образов.

Знаковую картину мира агента-онколога рассмотрим со стороны сетевой организации. Каузальная матрица представляет собой $z = (e_1, e_2, \dots, e_h)$ размерности q на h , где q - размерность входных событий, а h - длина последовательности входных событий. Каждое событие x может быть в данном случае представлено неким набором из q признаков, характеризующих данный знак. При этом каждый столбец e_i каузальной матрицы является бинарным вектором длины q и кодирует те признаки, которые должны присутствовать во входном событии в момент времени t .

Таблица 1. Семантические связи на множестве образов знака «Рак поджелудочной железы»

	Факторы риска	Симптомы	Клинические проявления
Тип семантической связи	Потенсивная связь	Комитативная связь	Комитативная связь

Табл. 2. Признаки знака «Рак поджелудочной железы»

	Расположение опухоли	Стадия	TNM-классификация
Тип свойства	Локатив	Квантитатив	Квалитатив

В нашем случае длина последовательности представляет собой замеры во времени, в течение которого диагностируют болезнь или протекает болезнь. В качестве временных рамок для образной компоненты могут выступать, например, количество осмотров у врача, рекомендованное Минздравом, или то количество осмотров, которое врач сделал. В каждый момент времени должны присутствовать признаки, необходимые для обнаружения рака поджелудочной железы. Для смысловой компоненты последовательность входных событий определяется количеством реальных осмотров у врача и разделяется на множество столбцов-условий и эффектов. В качестве эффектов выступают результаты диагностики. Например, соответствие поставленного диагноза реальному. Для разработки алгоритма принятия решения коллективом агентов будут

рассматриваться различия между образной компонентой, то есть неким эталонным описанием диагностики рака поджелудочной железы, и смысловой компонентой, то есть реальным опытом врача.

Ниже приведены примеры знаков картины мира агента-онколога. На Рис. 1 и Рис.2 изображены знаки «Поджелудочная железа» и «Рак поджелудочной железы», на которых схематично изображены каузальные матрицы с описанием их содержания.

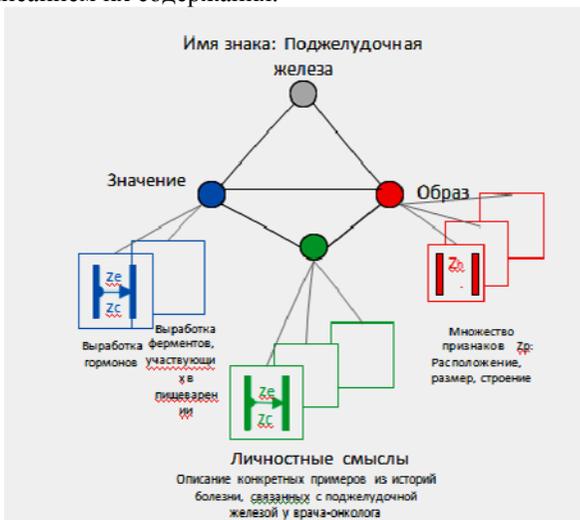


Рис.1 Знак «Поджелудочная железа»

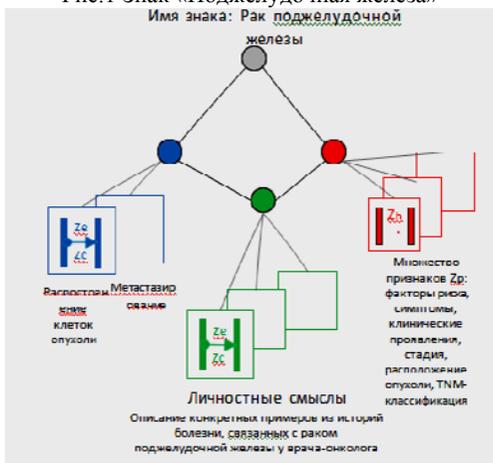


Рис. 2 Знак «Рак поджелудочной железы»

Для каждого знака построены каузальные матрицы, которые объединены в каузальную сеть. На Рис. 3 и Рис. 4 представлены некоторые примеры отношений на каузальной сети, моделирующие картину мира агента-онколога. На Рис. 4 приведены примеры отношения классификации и отношения «является элементом образа». К первому типу отношения относятся связи между узлами «Факторы риска РПЖ» и «Курением», «Ожирением», «Сахарным диабетом», между «Клиническими проявлениями» и «Компрессией», «Обтурацией», «Интоксикацией», между «Симптомами РПЖ» и «Механической желтухой», «Кожным зудом», «Лихорадкой», а также между каждой из стадий, описанный с помощью TNM-классификации, и самим этим понятием и каждой пронумерованной стадией и общим понятием стадии. Стоит отметить, что для упрощения модели каждый знак представлен только одной каузальной матрицей, а также на схеме были изображены для наглядности лишь некоторое подмножество всех возможных значений симптомов, факторов риска и названий стадий заболевания. Элементы «Курение», «Ожирение», «Сахарный диабет», «Компрессия», «Обтурация», «Интоксикация», «Механическая желтуха», «Кожный зуд», «Лихорадка» являются элементами образа «Рак поджелудочной железы».

На Рис. 3 изображен пример сценарного отношения и отношения классификации со знака «Компьютерная томография». Следует отметить, что данный пример является одним из примеров диагностики.

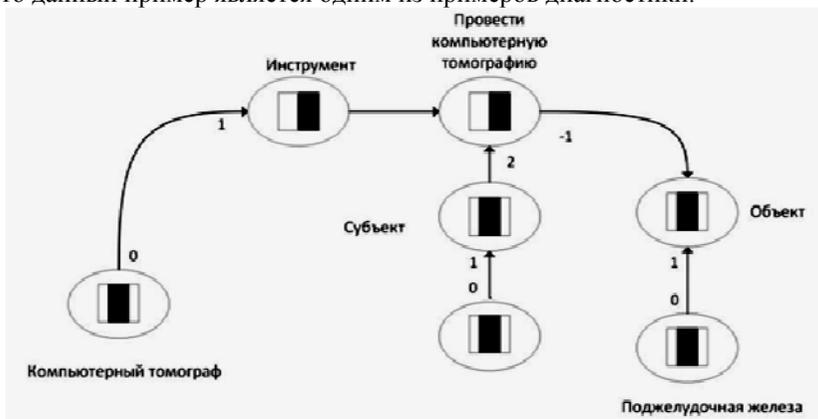


Рис.3 Пример сценарного отношения со знаком «Компьютерная томография»

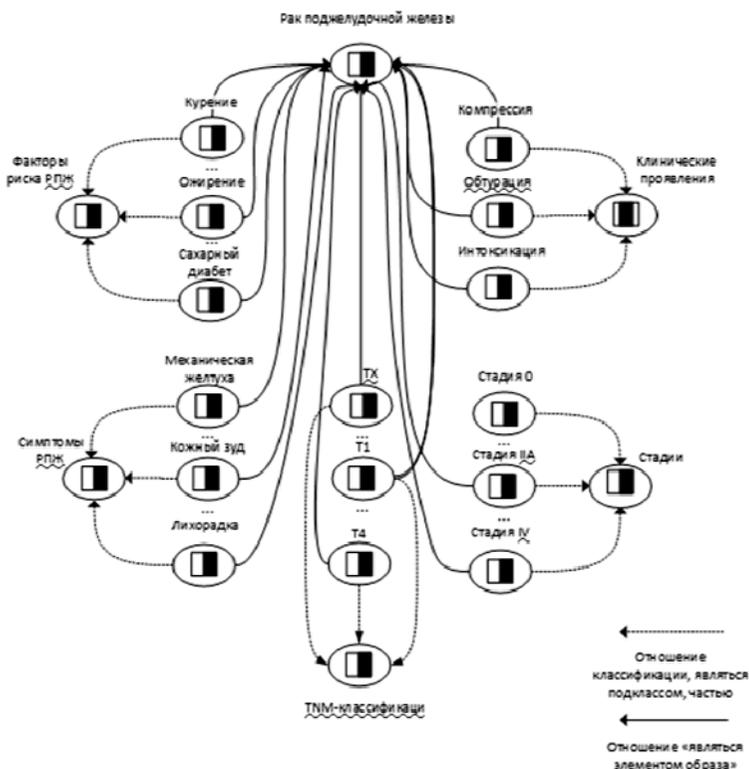


Рис.4 Пример отношений на каузальной сети образов

Заключение

В статье описана знаковая картина мира агента-онколога для решения задачи моделирования работы консилиума врачей при диагностике рака поджелудочной железы. Кроме онколога в консилиум входят радиолог, гастроэнтеролог, патологоанатом, ЛПР хирург. Картины мира других агентов построены аналогичным образом. На текущем этапе исследования ведется разработка алгоритма принятия решения коллективом агентов для диагностики рака поджелудочной железы.

Список литературы

[Panov A. I., et al. 2017] Panov A. I., Yakovlev K. Behavior and path planning for the coalition of cognitive robots in smart relocation tasks //Robot Intelligence Technology and Applications 4. – Springer, Cham, 2017. – С. 3-20.

[Kim J. H. et al 2015] Kim J. H. et al. Robot Intelligence Technology and Applications. – Springer, Cham, 2015.

[Albus J. S., 2002] Albus J. S. 4D/RCS: a reference model architecture for intelligent unmanned ground vehicles //Unmanned Ground Vehicle Technology IV. – International Society for Optics and Photonics, 2002. – Т. 4715. – С. 303-311.

[Yoo J. K., et al, 1984] Yoo J. K., Kim J. H. Gaze control-based navigation architecture with a situation-specific preference approach for humanoid robots //IEEE/ASME Transactions on Mechatronics. – 2015. – Т. 20. – №. 5. – С. 2425-2436.

[Rosenfeld A. et al, 2017] Rosenfeld A. et al. Intelligent agent supporting human-multi-robot team collaboration //Artificial Intelligence. – 2017. – Т. 252. – С. 211- 231.

[Chae S. W, 2015] Chae S. W., Seo Y. W., Lee K. C. Task difficulty and team diversity on team creativity: Multi-agent simulation approach //Computers in Human Behavior. – 2015. – Т. 42. – С. 83-92.

[Осипов Г. С. и др, 2017] Осипов Г. С. и др. Знаковая картина мира субъекта поведения //Москва: ФИЗМАТЛИТ. – 2017.

[Осипов Г. С. и др, 2014] Осипов Г. С., Панов А. И., Чудова Н. В. Управление поведением как функция сознания. I. Картина мира и целеполагание //Известия Российской академии наук. Теория и системы управления. – 2014. – №. 4. – С. 49-49

[Осипов Г. С, 2018] Осипов Г. Методы искусственного интеллекта. – Litres, 2018. С 197 -198

ИДЕНТИФИКАЦИЯ АККАУНТОВ ПОЛЬЗОВАТЕЛЕЙ В СОЦИАЛЬНЫХ СЕТЯХ «ВКОНТАКТЕ» И «ОДНОКЛАССНИКИ»¹

А.А. Корепанова

М.В. Абрамов (*mva@dscs.pro*)

Т.В. Тулупьева (*tvt@dscs.pro*)

Санкт-Петербургский государственный университет,
Санкт-Петербург

Санкт-Петербургский институт информатики и
автоматизации Российской академии наук, Санкт-
Петербург

В работе предоставлена методика сопоставления публичных анкет пользователей социальных сетей «ВКонтакте» и «Одноклассники». Предложены методы обработки данных и получения итоговой оценки схожести профилей. Работа может быть полезна для идентификации аккаунтов пользователя в рамках исследования социоинженерных атак.

Ключевые слова: социоинженерные атаки, нечеткое сравнение, информационная безопасность

Введение

По данным отчёта Ironscale за 2017 год, из всех успешных атак на информационные системы, более 90% пришлось на социоинженерные [IRONSCALES, 2017]. Это подтверждает, что одним из самых уязвимых мест информационной системы остаётся пользователь. Методы социальной инженерии обычно основаны на манипулятивных техниках, в связи с чем особый интерес для злоумышленников представляют психологические, социологические, культурантропологические и иные особенности пользователей, исходя из которых можно сделать

¹ Работа выполнена в рамках проекта по государственному заданию СПИИРАН № 0073-2019-0003 и при финансовой поддержке РФФИ (гранты №18-01-00626, № 18-37-00323)

предположения об их потенциальных уязвимостях. Информацию о ряде этих особенностей пользователя можно получить путём анкетирования, тестирования, специализированных игр и иными способами. В статье рассмотрим социальные сети в качестве источника информации о личностных особенностях пользователя. Важно то, что контент, извлекаемый из этого источника, как правило, доступен злоумышленнику, а его анализ зачастую позволяет получить больше информации, чем посредством психологических тестов или интервью [Du, 2018, Суворова, 2012]. Оценить степень выраженности личностных особенностей пользователя можно на основании анализа размещённых им постов, списка его друзей, подписок на группы и иные сообщества, личной информации, содержащейся в анкете. Уже существуют наработки в части построения профиля психологических особенностей пользователя на основе информации, извлекаемой из социальных сетей [Абрамов и др., 2018a]. На основе профиля психологических особенностей можно построить фрагмент профиля уязвимостей пользователя [Багрецов и др., 2017]. Оценки вероятности защищенности/поражаемости пользователей информационных систем строятся с опорой на профиль уязвимостей. Отметим, что чем больше информации о пользователе удаётся извлечь из различных источников, тем проще оценить степень выраженности его личностных особенностей и в дальнейшем — уязвимостей.

В России на данный момент пользуются наибольшей популярностью несколько социальных сетей, каждая из которых имеет свою направленность и специфику размещаемого контента. К таким ресурсам можно отнести «Youtube», «ВКонтакте», «Facebook», «Одноклассники», «Instagram» [WebCanape, 2018]. Многие интернет-пользователи имеют несколько аккаунтов в разных социальных сетях и публикуют в них различный или схожий контент. Таким образом, чем больше аккаунтов пользователя мы сможем найти, сопоставить и проанализировать, тем более правдоподобным и полным окажется построенный фрагмент профиля уязвимостей пользователя. В настоящее время отсутствуют эффективные, высокоточные, открытые инструменты для сопоставления аккаунтов пользователя в социальных сетях «ВКонтакте» и «Одноклассники». Такие разработки позволили бы агрегировать больше данных о пользователях для оценки степени выраженности их личностных особенностей и, опосредованно, уязвимостей. Материал данной статьи направлен на разрешение необходимости сопоставления различных аккаунтов пользователей в социальных сетях и поиск тех из них, которые принадлежат одному человеку для двух социальных сетей: «ВКонтакте» и «Одноклассники», пересечение аудитории которых по

некоторым оценкам составляет 19 млн пользователей [Hiconversion, 2018].

1. Релевантные работы

Данная работа является частью общего исследования, посвященного проблеме анализа защищенности пользователей информационных систем от социоинженерных атак, которое проводится на базе лаборатории теоретических и междисциплинарных проблем информатики СПИИРАН. В рамках этого исследования уже разработаны алгоритмы поиска сотрудников компании в социальных сетях [Shindarev et al., 2017], а также алгоритмы восстановления пропущенных значений метапрофиля пользователя [Абрамов, 2018], которые явились заделом для настоящей работы. Разработанный в соответствии с предложенным в данной статье алгоритмом программный модуль взаимодействует с упомянутыми уже разработанными.

Также исследованиями, ассоциированными с более общей задачей построения фрагмента профиля уязвимостей пользователей, являются, работы, основанные на агрегации данных о пользователе из социальных сетей [Суворова и др., 2012, Абрамов и др., 2018b, Тулупьева и др., 2010, Мальцеская и др., 2017, Bagretsov et al., 2017]. В них решаются задачи выявления степени выраженности психологических особенностей пользователя на основе анализа контента, публикуемого им на своей странице в социальной сети [Суворова и др., 2012, Абрамов и др., 2018b, Тулупьева и др., 2010, Мальцеская и др., 2017, Bagretsov et al., 2017]. Помимо этого, из социальных сетей можно извлекать информацию об интенсивности взаимодействия пользователей между собой, которая лежит в основе оценок вероятности успеха многоходовых социоинженерных атак [Абрамов и др., 2018b].

Отметим, что задача автоматизации идентификации пользователей с использованием информации, самостоятельно публикуемой пользователем о себе в социальных сетях и находящейся в открытом доступе (личная информация, содержащаяся в анкете в социальной сети, размещённые посты, списки друзей и т.п.) не нова [Gae-won et al., 2011, Motoyama et al., 2009, Raad et al., 2010, Vosecky et al., 2009, Kong et al., 2013, Li et al., 2018]. В исследованиях [Gae-won et al., 2011, Motoyama et al., 2009, Raad et al., 2010, Vosecky et al., 2009] предлагаются подходы к анализу публичных анкет пользователей социальных сетей «MySpace», «StudiVZ». Предлагаемые методы заключаются в построении векторов признаков, содержащихся в анкете пользователя. К полученным векторам применяются методы точного, частичного и нечёткого сравнения. По данным [WebCanape, 2018], социальные сети, «MySpace», «StudiVZ»,

которые анализируются в этих материалах, мало распространены в России. Авторами данных исследований были разработаны алгоритмы, показавшие эффективность около 80% на тестовой выборке аккаунтов, были предложены признаки, являющиеся наиболее существенными при сравнении аккаунтов. В работах [Raad at al., 2010, Vosecky at al., 2009] предлагаются методы идентификации аккаунтов пользователя посредством анализа публикуемого на их страницах текстового контента. В [Raad at al., 2010] представлены выводы о том, что автора поста можно определить по уникальному стилю письма. В [Vosecky at al., 2009] используется подход, согласно которому учитывается не только текстовый контент, публикуемый пользователем в посте, но и ассоциированная с ним метаинформация: геолокация, время публикаций, хештеги и т.п.

Таким образом, проблема автоматизированной идентификации аккаунтов пользователей в социальных сетях «ВКонтакте» и «Одноклассники» не решена. Её решение в рамках тематики социоинженерных атак позволило бы аккумулировать большее количество информации, способствующей оценкам степени выраженности личностных особенностей пользователя и, опосредованно, его уязвимостей.

2. Постановка задачи

Данное исследование посвящено решению задачи сопоставления аккаунтов пользователей в разных социальных сетях. В настоящей работе рассматриваются социальные сети «ВКонтакте» и «Одноклассники» как источники аккаунтов для сравнения. Параметрами для сравнения аккаунтов пользователя в разных социальных сетях будут выступать следующие его цифровые следы, выгружаемые посредством соответствующего программного обеспечения [Фильченков и др., 2014]: анкетные данные; социальное окружение, представленное списком друзей, родственников и семейным положением; ассоциированная с публикуемым контентом метаинформация, а именно геолокация, хештеги, время публикации. В результате реализации алгоритма ожидается получение значения оценки вероятности принадлежности двух аккаунтов в разных социальных сетях одному пользователю. Для получения данной оценки необходимо решить несколько независимых подзадач, состоящих в сравнении публичных анкет аккаунтов, поиске пересечения социального окружения, определения географической близости указанных у постов геолокаций, анализе статистики нахождения пользователя онлайн. Эта работа посвящена первой подзадаче – сопоставлению аккаунтов

пользователей на основе сравнения публичных анкет аккаунтов. Поля публичных анкет указанных социальных сетей в большинстве своем совпадают: в них содержится основная личная – об имени, поле, возрасте, городе проживания, родном городе, образовании и карьере пользователя – и контактная информация. Методика сопоставления публичных анкет аккаунтов для социальных сетей «ВКонтакте» и «Одноклассники» основана на сравнении соответствующих полей публичных анкет.

3. Методика сопоставления аккаунтов

Методика сопоставления аккаунтов состоит из трёх этапов: сбор анкетных данных, сравнение полученных данных и вычисление оценки схожести. Рассмотрим каждый шаг более подробно.

1.1. Анкетные данные

1.1.1. Выгрузка

В качестве входных данных выступают две ссылки на аккаунты в разных социальных сетях. С помощью специально разработанного программного обеспечения осуществляется выгрузка анкетных данных по заданным ссылкам. Поскольку в разных социальных сетях представление анкеты пользователя отличается, напрямую сравнивать их представляется невозможным или очень трудоемким. Например, во «ВКонтакте» для девичьей фамилии в анкете пользователя существует отдельное поле, а в «Одноклассниках» некоторые пользователи дописывают её в круглых скобках после основной фамилии. Поэтому по каждому аккаунту из исследуемых социальных сетей строится метапрофиль, представляющий собой совокупность унифицированных анкетных данных пользователя. Далее перечислены данные из анкеты пользователя, входящие в метапрофиль, с введенными обозначениями:

- имя;
- фамилия;
- девичья фамилия;
- пол;
- адрес страницы (доменное имя, ссылка на эту страницу);
- дата рождения;
- родной город;
- город проживания;
- карьерная история (места и время работы) в следующем формате: «наименование организации – город – дата начала работы – дата конца работы»;
- образование в формате «тип учебного заведения – наименование

учебного заведения – специализация/кафедра/факультет – город – дата начала обучения – дата конца обучения»;

- семейное положение;
- о себе;
- интересы;
- любимая музыка;
- любимое кино;
- любимые книги.

1.1.2. Предобработка

Предобработка данных необходима, чтобы исключить факторы, снижающие качество данных и мешающие работе алгоритмов сравнения. Подготовка данных для сравнения включает в себя перевод букв в нижний регистр во всех полях; обратную транслитерацию полей «имя», «фамилия», «девичья фамилия», если это необходимо; удаление из поля «интересы» небуквенных символов. Например, исходные поля имя и фамилия могут быть заданы как «IVANOV Vasya», «ИВАНОВ Vasya» и т.п., после предобработки будет модернизировано в «иванов васыа».

1.1.3. Унификация

Полученные данные необходимо адаптировать таким образом, чтобы они были пригодны для сравнения, так как форма их представления может быть различна в зависимости от сервиса. Так значение поля «пол», полученное через API «ВКонтакте», будет представлено цифрой от 0 до 2, а при обращении к API «Одноклассники» можно извлечь слово «male» или «female». Унифицировать будем следующие поля: «имя», «пол», «дата», «семейное положение». Ниже представлена специфика унификации для указанных полей.

1) «Имя» будет приводиться в полную форму с использованием словаря имён, созданного на основе данных сайта kugifin.ru в котором в соответствие поставлены полная и уменьшительные формы имени. Если соответствующая данному имени пара «полное имя» – «уменьшительная форма» в словаре отсутствует, будем сравнивать исходное имя методом Джаро-Винклера [Winkler, 1990] с содержащимися в словаре уменьшительными формами. Полная форма того имени, которое окажется наиболее близко, будем считать за полную форму исходного.

Табл 1. Пример структуры словаря.

Полная форма	Вариация
Алексей	Алёша
Алексей	Лёха
Евгений	Женя
Евгений	Жека

2) В зависимости от сервиса «пол» может быть получен в формате цифры или слова. Эти данные будем хранить в форме цифры: 1 – женский, 2 – мужской, 0 – не указан.

3) Дату можно получить из разных социальных сетей в двух форматах: «1990-11-29» или «29.11.1990». Для единообразия будем представлять дату во втором формате.

4) «Семейное положение» варьируется в разных социальных сетях: в «Одноклассниках» всего 5 вариантов заполнения этого поля, во «ВКонтакте» – 10. Запись одинаковых по смыслу вариантов может различаться: так «В браке» в «Одноклассниках» соответствует «Женат/замужем» во «ВКонтакте», поэтому данные будем хранить в форме кода, согласованно со следующей таблицей соответствия.

Табл 2. Таблица соответствия значений поля «семейное положение»

Код	Одноклассники	ВКонтакте
1	В отношениях	Есть друг/есть подруга Помолвлен/помолвлена В гражданском браке
2	Открыт/открыта для общения	В активном поиске Не женат/не замужем
3	В браке	Женат/замужем
4	Не указано Разведен/разведена	Не указано Влюблён/влюблена Всё сложно

1.2. Сравнение

Соответствующие поля метапрофилей будем сравнивать друг с другом. Несмотря на предобработку и унификацию выгружаемых данных, запись некоторых сопоставляемых полей всё ещё может варьироваться из-за опечаток, ошибок, изменений форм слов. В соответствии с этим, к полям, которые не допускают вариации, применим точное сравнение, а

при сравнении остальных полей обратимся к аппарату нечеткого сравнения и семантическому сравнению. Выделим следующие типы сравнения:

- Точное сравнение. Применяется для полей «пол», «дата рождения», «семейное положение», «имя». Результат этого сравнения будет булева функция, принимающая значения 0 – при несовпадении и 1 – при совпадении.

- Нечёткое сравнение. Применяется для полей, сравнение которых необходимо проводить с учетом возможных преобразований строк, связанных с различиями в представлении данных в социальных сетях, опечатками, изменениями форм слов и др. Такими полями являются, например, «фамилия», «девичья фамилия» и «псевдоним». Для нечеткого сравнения этих полей применим метрику Джаро-Винклера. Согласно исследованию [20], данная метрика показывает высокую эффективность при решении подобных задач. Также в данную категорию попадают поля «о себе», «интересы», «любимая музыка», «любимое кино», «любимые книги». Результат этого сравнения – число от 0 до 1.

- Google-сравнение. Данный метод применяется для полей, содержимое которых может быть лексикографически далеко, но семантически совпадать. Применяется для полей: «родной город», «город рождения», «карьера», «образование». Идея метода заключается в отправке запросов к поисковой системе Google и сравнении результатов выдачи. Если среди первых десяти ссылок выдачи есть совпадения, то будем считать, что значения исследуемых полей идентичны. Например, для поля город в одном метапрофиле содержится значение «Ленинград», в другом – «Санкт-Петербург». Направляем запросы «Ленинград город» и «Санкт-Петербург город» в Google, анализируем выдачу, если есть совпадения записываем 1.

Отдельно остановимся на сравнении полей «образование» и «карьера». Эти поля могут быть представлены как массивы значений, каждый элемент такого массива для поля «образование» состоит в свою очередь из следующих полей: «тип учебного заведения», «наименование учебного заведения», «специализация/кафедра/факультет», «город», «дата начала обучения», «дата конца обучения». В связи с этим сравнение двух элементов таких массивов будем производить следующим образом: на точное совпадение проверяем даты начала и конца обучения, остальные поля, если есть, сравниваем Google-сравнением. Результаты каждого сравнения делим на количество заполненных полей и складываем. Результат – число от 0 до 1. Аналогично для поля «карьера».

Подробнее опишем сравнение полей, которые могут быть

представлены как массивы значений, к ним относятся поля «карьера», «образование» (представление описано выше), «о себе», «интересы», «любимая музыка», «любимое кино», «любимые книги» (их представим как массивы слов). Сопоставление таких массивов будет происходить через сравнение каждого элемента первого массива с каждым элементом второго с использованием соответствующего метода сравнения. Если сходство двух элементов выше некоторого определённого порогового значения, элементы считаем совпавшими. Результат сравнения двух массивов вычислим делением количества совпавших элементов на размер наибольшего массива из сравниваемых.

Результат сравнения полей двух метапрофилей представим в векторном виде, поля будут соответствовать полям метапрофиля и содержать результат сравнения соответствующих полей.

1.3. Получение оценки схожести

Итоговое значение оценки вероятности принадлежности двух аккаунтов в разных социальных сетях одному пользователю p рассчитывается по следующей формуле:

$$p = \sum a_i p_i$$

где p_i – результат сравнения поля i , a_i – соответствующий весовой коэффициент. Сумма таких весовых коэффициентов должна быть равна 1. Способ определения этих коэффициентов – тема дальнейшего исследования.

Заключение

В статье предложена методика решения задачи сопоставления публичных анкет профилей в различных социальных сетях. В качестве источника информации рассматриваются аккаунты социальных сетей «Одноклассники» и «ВКонтакте». Предложенная методика закладывает основу для дальнейшей работы по проведению соответствующих экспериментов, выработке новых алгоритмов. Полученные результаты будут способствовать идентификации разных аккаунтов одного пользователя в разных социальных сетях с целью агрегации большего количества информации, используемой при построении профиля уязвимостей пользователя.

Список литературы

[Абрамов и др., 2018a] Абрамов М.В., Тулупьев А.Л., Тулупьева Т.В. Агрегирование данных из социальных сетей для восстановления фрагмента метапрофиля пользователя // В сборнике: Шестнадцатая Национальная конференция по искусственному интеллекту с международным участием КИИ-2018 Труды конференции: в 2-х томах. 2018.

[Абрамов и др., 2018b] Абрамов М.В., Тулупьев А.Л., Сулейманов А.А. Задачи анализа защищённости пользователей от социоинженерных атак: построение социального графа по сведениям из социальных сетей // Научно-технический вестник информационных технологий, механики и оптики. 2018. Т. 18. № 2.

[Багрецов и др., 2017] Багрецов Г.И., Шиндарев Н.А., Абрамов М.В., Тулупьева Т.В. Подходы к разработке моделей для анализа текстовой информации в профилях социальной сети в целях построения профиля уязвимостей пользователя // В сборнике: Международная конференция по мягким вычислениям и измерениям. 2017. Т. 1.

[Мальчевская и др., 2017] Мальчевская Е.А., Бирилло А.И., Харитонов Н.А., Золотин А.А. Развитие матрично-векторного подхода в алгоритмах локального априорного вывода в алгебраических байесовских сетях // Труды VII Всероссийской научно-практической конференции «Нечеткие системы, мягкие вычисления и интеллектуальные технологии». Санкт-Петербург, 2017. Т. 1.

[Суворова и др., 2012] Суворова А.В., Тулупьева Т.В., Тулупьев А.Л., Сироткин А.В., Пашенко А.Е. Вероятностные графические модели социально-значимого поведения индивида, учитывающие неполноту информации // Труды СПИИРАН. 2012. Т. 3. № 22.

[Тулупьева и др., 2010] Тулупьева Т.В., Тулупьев А.Л., Пашенко А.Е., Азаров А.А., Степашкин М.В. Социально-психологические факторы, влияющие на степень уязвимости пользователей автоматизированных информационных систем с точки зрения социоинженерных атак // Труды СПИИРАН. 2010. Т. 1. № 12.

[Фильченков и др., 2014] Фильченков А.А., Азаров А.А., Абрамов М.В. Компаративный анализ методологических основ задач прогнозирования исходов выборов и протестов по цифровым следам пользователей в социальных медиа // В сборнике: Технологии информационного общества в науке, образовании и культуре сборник научных статей. Труды XVII Всероссийской объединенной конференции «Интернет и современное общество» (IMS-2014). 2014.

[Bagretsov et al., 2017] Bagretsov G.I., Shindarev N.A., Abramov M.V., Tulupyeva T.V. Approaches to development of models for text analysis of information in social network profiles in order to evaluate user's vulnerabilities profile // Proc. 20th IEEE Int. Conf. on Soft Computing and Measurements. St. Petersburg, 2017.

[Du, 2018] Du J., Jiang C., Chen K.C., Ren Y., Poor H.V. Communitystructured evolutionary game for privacy protection in social networks // IEEE Transactions on Information Forensics and Security. 2018. V. 13. N 3. doi: 10.1109/TIFS.2017.2758756

[Gae-won at al., 2011] Gae-won Y., Seung-won H., Zaiqing N., Ji-Rong W. SocialSearch: Enhancing Entity Search with Social Network Matching // EDBT/ICDT: Proceedings of the 14th International Conference on Extending Database Technology.

2011. doi: 10.1145/1951365.1951428

[Hiconversion, 2018] Статистика социальных сетей в России на 2018 год. <https://hiconversion.ru/blog/statistika-socialnyh-setej-v-rossii-na-2018-god/>

[IRONSCALES, 2017] New Email Security Report from IRONSCALES Identifies Email Phishing Attack Detection, Mitigation and Remediation as Biggest Challenge for Security Teams: <http://www.prweb.com/releases/2017/09/prweb14742215.htm>

[Kong at al., 2013] Xiangnan Kong, Jiawei Zhang, Philip S. Yu. Inferring anchor links across multiple heterogeneous social networks. // Proceedings of the 22nd ACM international conference on Information & Knowledge Management. 2013.

[Li at al., 2018] Yongjun Li, Zhen Zhang, You Peng, Hongzhi Yin, Quanqing Xu. Matching user accounts based on user generated content across social networks // Future Generation Computer Systems. 2018. Volume 83.

[Motoyama at al., 2009] Motoyama, M., Varghese, G. I Seek You - Searching and Matching Individuals In Social Networks // WIDM '09: Proceeding of the eleventh international workshop on Web information and data management. 2009. doi: 10.1145/1651587.1651604

[Raad at al., 2010] Raad, E., Chbeir, R., Dipanda, A. User Profile Matching in Social Networks // 13th International Conference on Network-Based Information Systems (NBIS). 2010.

[Shindarev at al., 2017] Shindarev N., Bagretsov G., Abramov M., Tulupyeva T., Suvorova A. Approach to identifying of employees profiles in websites of social networks aimed to analyze social engineering vulnerabilities // Advances in Intelligent Systems and Computing. 2017. V. 679. doi: 10.1007/978-3-319-68321-8_45

[Vosecky at al., 2009] Vosecky, J., Dan Hong, Shen, V.Y. User identification across multiple social networks // Proceedings of First International Conference on Networked Digital Technologies. 2009.

[WebCanape, 2018] Социальные сети в 2018 году: глобальное исследование. <https://www.web-canape.ru/business/socialnye-seti-v-2018-godu-globalnoe-issledovanie/>

[Winkler, 1990] Winkler, W. E. (1990). String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. Proceedings of the Section on Survey Research Methods. American Statistical Association.

МАЙНИНГ МНОЖЕСТВ ПРИЗНАКОВ НА ОСНОВЕ СЖАТИЯ: ВЕРОЯТНОСТНЫЙ ПОДХОД¹

Т.П. Махалова (*tpmakhalova@hse.ru*)
Национальный исследовательский университет
Высшая школа экономики, Москва
LORIA (CNRS – Inria – Университет Лотарингии),
Вандевр-ле-Нанси, Франция

С.О. Кузнецов (*skuznetsov@hse.ru*)
Национальный исследовательский университет
Высшая школа экономики, Москва

А. Наполи (*amedeo.napoli@loria.fr*)
LORIA (CNRS – Inria – Университет Лотарингии),
Вандевр-ле-Нанси, Франция

Майнинг паттернов (поиск множеств признаков в данных) является активно развивающимся направлением интеллектуального анализа данных. Большая часть современных подходов к поиску паттернов основана на принципе минимальной длины описания. Данный класс методов основан на оптимальном кодировании и подразумевает задание вероятностного пространства на множестве паттернов. В данной работе мы рассматриваем существующий подход к определению вероятностного пространства и предлагаем его модификацию, позволяющую уменьшить избыточность оптимального множества паттернов.

Ключевые слова: замкнутые понятия, минимальная длина описания, отбор множеств признаков

Введение

Майнинг паттернов (МП) занимает важное место среди методов обнаружения знаний и майнинга данных. Основной задачей МП является обнаружение небольшого набора интересных паттернов, обладающих в совокупности минимальной избыточностью, описывающих достаточно

¹ Работа выполнена при финансовой поддержке РФФ (проект № 17-11-01294).

полно анализируемые данные и предоставляющих возможность их интерпретации.

Все подходы к майнингу паттернов можно разделить на две группы: статические и динамические. Статические подходы подразумевают использование мер оценки интересности паттернов [Geng et al., 2006]. Поиск паттернов осуществляется при неизменных предположениях об интересности, например, поиск частых паттернов с частотой выше заданного порога. К основным недостаткам данного подхода относят следующие. Во-первых, вместо поиска “интересного” набора паттернов в совокупности, методы этой группы направлены на поиск отдельных интересных паттернов, не зависимо друг от друга. Такие паттерны, как правило, очень похожи и описывают избыточно анализируемые данные. Во-вторых, выбор мер интересности крайне субъективен. Зачастую, достаточно сложно обосновать выбор меры (в большинстве случаев, эксперт интуитивно выбирает ту или иную меру).

Динамические методы лишены перечисленных недостатков. Большинство динамических подходов основано на принципе минимальной длины описания (МДО) [Vreeken et al., 2011; Siebes et al., 2011; Smets et al., 2012]. Он опирается на предположение о том, что оптимальная модель обеспечивает максимальное сжатие данных. Применительно к МП, задача состоит в поиске модели (набора паттернов), минимизирующей двухступенчатый код $L(M) + L(D|M)$, где $L(M)$ – длина модели M и $L(D|M)$ – длина данных, закодированных с использованием данной модели. Минимизация данной целевой функции подразумевает два этапа: майнинг паттернов (редуцирование пространства поиска) и отбор МДО-оптимальных среди них. Модель M представлена как двухколонная кодовая таблица, где в первой колонке содержатся паттерны, а во второй – соответствующие им коды. Определение длин кодов паттернов, содержащихся в кодовой таблице, основано на введенном вероятностном распределении. Используемые вероятностные модели подвержены влиянию эвристик. Последнее делает модели плохо интерпретируемыми и толерантными к избыточности паттернов. В данной работе мы предлагаем новую вероятностную модель, основанную на частотных оценках. В экспериментах мы показываем, что данная модель менее подвержена влиянию эвристик и позволяет получить множества паттернов с меньшей избыточностью.

1. Основные понятия

Мы рассматриваем «транзакционные базы данных», представленные в виде бинарных таблиц. Фрагмент такой базы данных приведен на Рис. 1.

Паттерны в таких данных представляют собой множества признаков. Мы исследуем замкнутые множества, поскольку последние (а) являются максимальными множествами, которые включают паттерны одинаковой частоты, (б) позволяют представить любой из таких паттернов без потерь. Основные понятия, связанные с замкнутыми множествами признаков, приведены в терминах Анализа формальных понятий [Ganter et al., 1999].

1.1. Анализ формальных понятий. Основные понятия

Пусть задано множество объектов $G = \{g_1, g_2, \dots, g_n\}$, множество признаков $M = \{m_1, m_2, \dots, m_k\}$ и бинарное отношение между ними $I \subseteq G \times M$, тогда формальным контекстом называется тройка (G, M, I) . На множестве объектов и множестве признаков задана операция $(\cdot)'$. Для произвольных подмножеств $X \subseteq G$ и $Y \subseteq M$ она принимает следующий вид:

$$Y' = \{m \in M \mid \forall g \in Y : gIm\}, X' = \{g \in G \mid \forall m \in X : gIm\}.$$

X' представляет собой множество признаков, общих для всех объектов множества X , Y' – множество объектов, обладающих всеми признаками из Y . Формальным понятием называется пара (X, Y) , где $X \subseteq G$, $Y \subseteq M$ и $X' = Y$, $Y' = X$. X и Y называют объемом и содержанием формального понятия, соответственно. Произвольное множество $Z \subseteq M$ называют паттерном [Pasquier et al., 1999]. Размером паттерна Z называют мощность его содержания, т.е., $|Z|$, частотой паттерна называют размер объема соответствующего ему формального понятия, т.е. $|Z'|$.

Пример. Рассмотрим формальный контекст, представленный на Рис. 1. Множество формальных понятий, размер содержания и объема которых больше 1, составляют понятия $(g_1 g_4 g_5, AC)$, $(g_2 g_3 g_4, DE)$, $(g_2 g_4, CDE)$, $(g_1 g_2, BC)$. Данные понятия упорядочены по частоте (по убыванию, \downarrow), длине (\downarrow) и лексикографически (\uparrow).

<p>(a)</p> <table style="border-collapse: collapse;"> <tr><td>ABC</td></tr> <tr><td>BCDE</td></tr> <tr><td>DE</td></tr> <tr><td>ACDE</td></tr> <tr><td>AC</td></tr> </table>	ABC	BCDE	DE	ACDE	AC	<table style="border-collapse: collapse; border-top: 1px solid black; border-bottom: 1px solid black;"> <tr><th></th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th></tr> <tr><th>g_1</th><td>x</td><td>x</td><td>x</td><td></td><td></td></tr> <tr><th>g_2</th><td></td><td>x</td><td>x</td><td>x</td><td>x</td></tr> <tr><th>g_3</th><td></td><td></td><td></td><td>x</td><td>x</td></tr> <tr><th>g_4</th><td>x</td><td></td><td>x</td><td>x</td><td>x</td></tr> <tr><th>g_5</th><td>x</td><td></td><td>x</td><td></td><td></td></tr> </table>		A	B	C	D	E	g_1	x	x	x			g_2		x	x	x	x	g_3				x	x	g_4	x		x	x	x	g_5	x		x			<table style="border-collapse: collapse;"> <tr><td>A</td><td>B</td><td>C</td><td></td><td></td></tr> <tr><td></td><td>B</td><td>C</td><td>D</td><td>E</td></tr> <tr><td></td><td></td><td></td><td>D</td><td>E</td></tr> <tr><td>A</td><td></td><td>C</td><td>D</td><td>E</td></tr> <tr><td>A</td><td></td><td>C</td><td></td><td></td></tr> </table>	A	B	C				B	C	D	E				D	E	A		C	D	E	A		C			<table style="border-collapse: collapse; border-top: 1px solid black; border-bottom: 1px solid black;"> <tr><th>$I \cup -\log_2(\cdot)$</th><th></th></tr> <tr><td>A 3</td><td>3/15</td></tr> <tr><td>B 2</td><td>2/15</td></tr> <tr><td>C 4</td><td>4/15</td></tr> <tr><td>D 3</td><td>3/15</td></tr> <tr><td>E 3</td><td>3/15</td></tr> <tr><td>$\sum_{u \in U} u$</td><td>= 15</td></tr> </table>	$I \cup -\log_2(\cdot)$		A 3	3/15	B 2	2/15	C 4	4/15	D 3	3/15	E 3	3/15	$\sum_{u \in U} u$	= 15
ABC																																																																																			
BCDE																																																																																			
DE																																																																																			
ACDE																																																																																			
AC																																																																																			
	A	B	C	D	E																																																																														
g_1	x	x	x																																																																																
g_2		x	x	x	x																																																																														
g_3				x	x																																																																														
g_4	x		x	x	x																																																																														
g_5	x		x																																																																																
A	B	C																																																																																	
	B	C	D	E																																																																															
			D	E																																																																															
A		C	D	E																																																																															
A		C																																																																																	
$I \cup -\log_2(\cdot)$																																																																																			
A 3	3/15																																																																																		
B 2	2/15																																																																																		
C 4	4/15																																																																																		
D 3	3/15																																																																																		
E 3	3/15																																																																																		
$\sum_{u \in U} u$	= 15																																																																																		
(a)	(b)	(c)	(d)																																																																																

Рис.1. Транзакционная база данных (а), ее представление в виде формального контекста (b) и представление, используемое в данной статье (с). Стандартная кодовая таблица ST (d), $L(D, ST) = 46.1$, см. детали в Разделе 1.2.

1.2. Принцип минимальной длины описания в задаче отбора паттернов

В контексте МДО, под оптимальным набором паттернов понимается такое множество паттернов, кодирование на основе которых обеспечивает максимальное сжатие данных [Grünwald, 2007].

Основу данного подхода составляет кодовая таблица, в которую записаны некоторые паттерны и длины их кодов в битах. Процесс майнинга паттернов состоит в минимизации длины $L(CT, D) = L(D|CT) + L(D|CT)$, где $L(D|CT)$ – длина набора данных D , закодированных с помощью кодовой таблицы CT , $L(CT|D)$ – длина кодовой таблицы CT , вычисленной на основе набора данных D . Далее, для краткости, мы будем обозначать множество паттернов из кодовой таблицы как CT . Кодирование объекта заключается в нахождении непересекающихся паттернов в CT , которые полностью покрывают описание данного объекта. Для нахождения покрытия паттерны таблицы просматриваются в фиксированном порядке. Изначально, за множество непокрытых признаков принимается полное описание объекта. Далее, для каждого последующего паттерна проверяется, содержит ли данный паттерн еще не покрытое описание. Если паттерн содержится в непокрытом описании, он добавляется в результирующее множество-покрытие а его признаки исключаются из множества непокрытых признаков описания. Как только множество непокрытых признаков оказывается пустым, функция возвращает множество паттернов – покрытие описания. Данная функция обозначается далее как $cover(\{g\}', CT)$. Количество использований паттерна X для покрытия объектов в D обозначается $usage(X) = |\{g \in G \mid X \in cover(\{g\}', CT)\}|$. Вероятность паттерна при фиксированной схеме кодирования вычисляется по следующей формуле: $P(X) = usage(X) / \sum_{X^* \in CT} usage(X^*)$.

Для определения длины кодов паттернов используется кодирование Шеннона, т.е. $L(code(X)) = -\log(P(X))$, обеспечивающее оптимальные длины кодов в данной вероятностной модели [Grünwald, 2007]. Длина набора данных D , закодированных с использованием CT , вычисляется по формуле $L(D|CT) = \sum_{X \in CT} usage(X)L(code(X))$. Поскольку майнинг паттернов не преследует цели кодирования данных, в качестве целевой функции используется упрощенная длина данных и кодовой таблицы (допускается вещественное количество бит и опускается кодирование непосредственно паттернов – правого столбца кодовой таблицы).

Кодовая таблица, где все паттерны являются одиночными признаками,

называется стандартной кодовой таблицей и обозначается ST (см. Рис. 1, d).

Пространство поиска оптимального набора паттернов представляет собой множество всех возможных подмножеств различных паттернов. Все возможные паттерны, в свою очередь, представляют булеан множества признаков M . Таким образом, размер пространства поиска 2^N , где $N = 2^{|M|}$. На практике пространство поиска ограничивается подмножеством паттернов, которое называют множеством кандидатов. Для выбора множества кандидатов применяют различные эвристики [Поспелов и др. 1967; Гладун, 1977; Ройзензон, 2005].

1.2.1. Принципы вычисления кодовых таблиц

На начальном этапе кодовая таблица состоит из одноэлементных паттернов $\{m \mid m \in M\}$. Также имеется набор кандидатов – множество паттернов, упорядоченное в соответствии с выбранной мерой. Минимизация длины заключается в последовательном добавлении лучшего (по выбранной мере) паттерна, перевычислении покрытия с обновленным набором паттернов и вычисления новой длины. Если новая длина короче предыдущей, то данный паттерн добавляется в кодовую таблицу. Процесс продолжается до тех пор, пока все паттерны из набора кандидатов не будут просмотрены.

Стандартным порядком кандидатов называют упорядочивание по частоте (\downarrow), длине (\downarrow) и лексикографически (\uparrow) [Vreeken et al., 2011]. Паттерны в самой кодовой таблице могут быть упорядочены в соответствии с другими мерами. Так упорядочивание паттернов по длине (\downarrow), частоте (\downarrow) и лексикографически (\uparrow) называют стандартным порядком покрытия. Именно в этом порядке паттерны используются при жадном покрытии данных.

Одним из наиболее распространенных МДО-подходов к майнингу паттернов является Krimp [Vreeken et al., 2011].

Пример. Рассмотрим принцип работы Krimp на фрагменте данных, приведенном на Рис. 1. На начальном этапе рассматривается стандартная кодовая таблица (Рис. 1, d). Далее, кандидаты добавляются в таблицу в стандартном порядке кандидатов. Кандидаты в таблице упорядочиваются в соответствии со стандартным порядком покрытия. Если покрытие текущим набором паттернов обеспечивает меньшую длину, паттерн-кандидат принимается в таблицу.

$C - \log_2(\cdot)$											
ABC	AC	3/12	ABC	AC	3/12	ABC	CDE	2/12	ABC	C	$-\log_2(\cdot)$
BCDE	B	2/12	BCD	DE	3/12	BCDE	AC	2/12	BCDE	AC	3/8
DE	C	1/12	D	B	2/12	DE	DE	1/12	DE	DE	3/8
A CDE	E	3/12	A CD	C	1/12	A CDE	A	1/12	A CDE	BC	1/8
A C	D	3/12	A C			A C	B	2/12	A C	B	1/8
(a)	(b)		(c)	(d)		(e)	(f)		(g)	(h)	

Рис. 2. Шаги Krimp. Кандидаты упорядочены по частоте (\downarrow), длине (\downarrow), лексикографически. Паттерны в таблице упорядочиваются по длине (\downarrow), частоте (\downarrow), лексикографически (\uparrow). Шаг 1: добавление AC (a-b), $L(D, CT)=39.0 < L(D, ST)$, паттерн принимается в таблицу. Шаг 2: добавление DE (c-d), $L(D, CT)=25.5 < 39.0$, паттерн принимается. Шаг 3: добавление CDE (e-f), $L(D, CT) = 30.0 > 25.5$, паттерн удаляется. Шаг 3: добавление BC (g-h), $L(D, CT)=23.3 < 25.5$, паттерн принимается.

2. Вероятностные модели для кодирования паттернов

В предыдущем разделе было отмечено, что определение кодов на основе вероятностного распределения обеспечивает оптимальную длину кодирования. Однако в теории сжатия информации, в частности применительно к МП, вопросу определения вероятностных пространств не было уделено должного внимания. В данном разделе мы рассматриваем вероятностную модель, используемую в методах майнинга паттернов на основе МДО, и ее недостатки, а также вводим новую модель, позволяющую минимизировать эти недостатки.

2.1. Модель на основе функции покрытия

Во всех существующих подходах к МП на основе МДО, вероятностное пространство определяется в соответствии с алгоритмом жадного покрытия. Другими словами, вероятность паттерна зависит от вероятности других паттернов, которые ранее использовались при покрытии данных.

Пусть $X_1, X_2, \dots, X_n \in CT$ множество упорядоченных паттернов кодовой таблицы, $n(X_i)$ – количество объектов, описание которых содержит паттерн X_i ; $n(X_i \bar{X}_j)$ – количество объектов, описание которых содержит паттерн X_i и не содержит паттерн X_j . Тогда $usage(X_i) = n(X_i \bar{X}_{i_1} \dots \bar{X}_{i_k})$, где $i_1, \dots, i_k < i$, $\bar{X}_{i_j} \cap X_i \neq \emptyset$, $usage(X_i) \leq frequency(X_i)$, что следует из рассмотренной стратегии жадного покрытия. Определенное таким образом вероятностное пространство подразумевает учет достаточно сложной зависимости между паттернами, а также покрытие данных без

пересечений. В экспериментах мы покажем, что это приводит к оптимистичным оценкам избыточного описания.

2.2. Модель на основе независимых паттернов

В данной работе мы предлагаем вероятностную модель, основанную на предположении о независимости паттернов в кодовой таблице, т.е. вероятность наблюдения паттерна X в данных вычисляется независимо от других паттернов из CT . В результате, оценки вероятности оцениваются не на основе частоты встречаемости в покрытии, а на основе частоты встречаемости в данных, т.е. $P(X) = frequency(X) / \sum_{X^* \in CT} frequency(X^*)$. Предположение о независимости паттернов облегчает интерпретацию модели и снижает зависимость оценок длин от способа покрытия данных (порядка паттернов в кодовой таблице).

3. Эксперименты

В данном разделе мы приводим результаты экспериментов, показывающие, что замена оценок позволяет существенно уменьшить избыточность описания и размер МДО-оптимального набора паттернов. Мы использовали 20 наборов данных из репозитория LUCS-KDD [Coenen, 2003].

Существует большое количество различных мер оценки качества множества признаков [Ignatov et al., 2015]. Условно их можно разделить на 3 группы: те, что оценивают простоту анализа (интерпретации), избыточность (разнообразие) и описательную способность.

В данной работе мы выбрали указанные ниже меры.

– *Количество паттернов ($|CT|$)* в таблице. Меньшее количество предпочтительнее, поскольку облегчает последующий анализ паттернов.

– *Коэффициент перекрытия (O)*, среднее число паттернов, приходящихся на покрытые отношения «объект-признак». Оценивает избыточность набора паттернов. Наборам паттернов с отсутствием избыточности соответствует коэффициент перекрытия 1.

– *Коэффициент непокрытия (U)*, доля непокрытых отношений «объект-признак». Оценивает описательную способность паттернов. Значения близкие к 0 соответствуют множествам паттернов с наилучшей описательной способностью.

Табл. 1

Порядок кандидатов	$ X ^* X' \downarrow, X' \downarrow,$ лексикограф. \uparrow				$ X' \downarrow, X \downarrow,$ лексикограф. \uparrow			
Порядок покрытия	$ X ^* X' \uparrow,$ лексикогр. \uparrow		$ X \downarrow, X' \downarrow,$ лексикогр. \uparrow		$ X ^* X' \uparrow,$ лексикогр. \uparrow		$ X \downarrow, X' \downarrow,$ лексикогр. \uparrow	
Оценки	<i>us.</i>	<i>freq.</i>	<i>us.</i>	<i>freq.</i>	<i>us.</i>	<i>freq.</i>	<i>us.</i>	<i>freq.</i>
$L(D,CT)/L(D,ST)$	0,59	0,69	0,53	0,70	0,66	0,78	0,53	0,78
$ CT $	29,23	20,91	123,82	21,41	50,14	20,05	114,82	20,14
O	2,75	2,12	12,58	2,26	3,74	1,69	11,37	1,71
U	0,32	0,26	0,20	0,26	0,22	0,25	0,19	0,25

В результате серии экспериментов, проведённых с разным упорядочиванием кандидатов и паттернов (покрытия), было выявлено, что предложенные оценки вероятности (столбец «*freq.*»), позволяют получить кодовые таблицы меньшего размера (строка « $|CT|$ »). В случае, когда используется стандартный порядок покрытия (столбец « $|X|\downarrow, |X'|\downarrow,$ лексикограф. \uparrow »), размер таблиц уменьшается более чем в 5 раз. Кроме того, значительно уменьшается избыточность наборов паттернов (строка « O »), при этом описательная способность сокращается лишь на 4,5% (строка « U »). В случае упорядочивания кандидатов по « $|X|^*|X'|\downarrow, |X'|\downarrow,$ лексикограф. \uparrow » и покрытия по « $|X|^*|X'|\uparrow,$ лексикограф. \uparrow », описательная способность также улучшается вместе с уменьшением избыточности и размера кодовых таблиц.

Стоит отметить, что коэффициент компрессии $L(D,CT)/L(D,ST)$ повышается лишь незначительно. Данный параметр не свидетельствует об ухудшении модели. Экспериментально было показано, что для одних и тех же наборов паттернов, коэффициент компрессии в рамках предложенной вероятностной модели всегда выше.

Заключение

В данной статье была предложена новая вероятностная модель для методов майнинга паттернов на основе принципа минимальной длины описания. В результате экспериментов было показано, что введенные оценки позволяют получить более компактные наборы признаков, обладающих меньшей избыточностью и сохраняющих при этом описательную способность (в сравнении с множествами признаков большего размера).

Список литературы

- [Coenen, 2003] Coenen F. The lucs-kdd discretised/normalised arm and carm data library – <http://www.csc.liv.ac.uk/~frans/KDD/Software/LUCS KDD DN>
- [Ganter et al., 1999] Ganter B., Wille R. Formal concept analysis: Logical foundations
- [Ignatov et al., 2015] Ignatov D.I., Gnatyshak D.V., Kuznetsov S.O., Mirkin B.G. Triadic formal concept analysis and triclustering: searching for optimal patterns // Machine Learning – Т. 101, №. 1-3, 2015..
- [Geng et al., 2006] Geng L., Hamilton H. J. Interestingness measures for data mining: A survey //ACM Computing Surveys (CSUR), 38(3):9, 2006.
- [Grünwald, 2007] Grünwald P. D. The minimum description length principle // MIT press. 2007
- [Pasquier et al., 1999] Pasquier N., Bastide Y., Taouil R., and Lakhal L. Efficient mining of association rules using closed itemset lattices // Information systems, 24(1):25–46, 1999.
- [Siebes et al., 2011] Siebes A., Kersten R. A structure function for transaction data // In Proceedings of SIAM, pages 558–569, 2011.
- [Smets et al, 2012] Smets K., Vreeken J. Slim: Directly mining descriptive patterns // In Proceedings of SIAM, pages 236–247, 2012.
- [Vreeken et al., 2011] Vreeken J., Van Leeuwen M., Siebes A. Krimp: mining itemsets that compress //Data Mining and Knowledge Discovery. 2011. №23 (1).
- [Гладун, 1977] Гладун В. П. Эвристический поиск в сложных средах. Киев: Наукова думка, 1977.
- [Поспелов и др., 1967] Поспелов Д. А., Пушкин В. Н., Садовский В. Н. Эвристическое программирование и эвристика как наука // Вопросы философии. 1967. №7. С. 45–56.
- [Ройзензон, 2005] Ройзензон Г. В. Способы снижения размерности признакового пространства для описания сложных систем в задачах принятия решений // Новости искусственного интеллекта. № 1, 2005.

УДК 004.043

ПРИМЕР РЕАЛИЗАЦИИ РЕКОМЕНДАТЕЛЬНОЙ СИСТЕМЫ НА ОСНОВЕ НЕЯВНЫХ ОЦЕНОК

С.Ю. Петрова (*mrs.lana@inbox.ru*)
Севастопольский государственный университет,
Севастополь

М.А. Бойкова (*sevostyanovamaria@gmail.com*)
Государственное учреждение «Новгородский
информационно-аналитический центр», Великий
Новгород

В работе описывается пример реализации рекомендательной системы на базе неявных оценок, использующей потоковую обработку данных Spark в сочетании с возможностями онлайн-обучения модели, основанными на SGD-методах MLlib Apache.

Ключевые слова: машинное обучение, потоковая обработка данных, неявные оценки, рекомендательные системы

Введение

Мировое развитие интернет торговли привело к активному развитию различного рода рекомендательных систем, которые помогают пользователям найти интересные для них товары, при этом, не тратя на это много времени. Существует огромное количество сервисов, которые рекомендуют пользователям самые различные элементы: музыку, фильмы, книги, видео, новости. Наиболее известные примерами подобных рекомендательных систем это Amazon, Ebay и Aliexpress. Принцип выработки рекомендаций этих сайтов основан на истории поисковой работы пользователя и истории прошлых покупок. Выигрыш от использования рекомендательных систем получают как сами пользователи, так и поставщики рекомендательных сервисов, так как эффективная рекомендательная система способна увеличить доходы компании и облегчить взаимодействие пользователей в онлайн-сообществах.

Общей задачей рекомендательных систем является повышение

качества обслуживания клиентов с помощью персонализированных рекомендаций, основанных на предварительных явных и неявных оценках. Числовое значение рекомендации с явной обратной связью указывает на предпочтение пользователей, тогда как числовое значение рекомендации с неявной обратной связи указывает на достоверность рекомендации. Рекомендательные системы, базирующиеся на явных оценках, за частую используют усредненный рейтинг товара или услуги, который был получен из оценок товаров, поставленных пользователями. Такой подход обладает рядом важных практических преимуществ. Он дает возможность работать со сравнительно простыми моделями исходных данных. Например, в контентных алгоритмах вполне допустимо ограничиваться таблицами типа «признак-значение признака». В рамках этого подхода предполагается, что каждый объект отождествляется с некоторой точкой многомерного признакового пространства, класс объектов представляется соответственно компактным множеством таких точек и задача распознавания сводится к построению детерминированного или вероятностного решающего правила, относящего новые объекты (точки) к одному из классов непересекающихся областей признакового пространства. Основная особенность этого подхода с точки зрения описания объектов фильтрации состоит в том, что в качестве формальных описаний — моделей объектов — используются множества, образованные признаками объектов; отношения между признаками, даже если они известны, в состав этих множеств, как правило, не входят. Также явные отзывы не всегда доступны и в таких системах имеется неразрешимая проблема «холодного старта» — это когда новый товар не имеет рейтинговой оценки.

Таким образом, предпочтительным методом является выработка рекомендации на базе неявных оценок. Эти системы пассивно отслеживают различные виды действия пользователей, например, такие как просмотр товара, работа с корзиной, оформление заказа и покупка товара. История покупок и повторяющееся поведение пользователя при работе с сайтом позволяет разработать модель пользовательских предпочтений. Анализ данных, полученных с помощью неявного подхода достаточно сложен, так как поведение пользователей не является детерминированным. Так если пользователь часто совершает одно и то же действие с определенным товаром, то это не указывает на его более высокое предпочтение. Одноразовое событие может быть вызвано различными причинами, которые не имеют ничего общего с пользовательскими предпочтениями, например, пользователь может посмотреть фильм один раз, или каждую неделю смотреть сериал, при

этом и фильм и сериал оценивать одинаково хорошо. Однако числовое значение неявной оценки определено полезно, так как оно говорит нам об уверенности, которую мы имеем в определенном наблюдении.

2. Обоснование выбора алгоритма фильтрации

Основным компонентом рекомендательной системы является алгоритм фильтрации. Алгоритм преобразует предварительно структурированный входной набор данных в матрицу рейтингов. Модели с латентным фактором включают в себя: Latent factor model (LFM), neural networks, latent Dirichlet allocation, matrix factorization. Поскольку рекомендательные модели Spark на основе факторизации матрицы демонстрируют отличную производительность в коллаборативной фильтрации [Koren Y., etc 2009], мы сосредоточили наше внимание на этом классе моделей. Целью коллаборативной фильтрации на основе факторизации является выявление скрытых особенностей данных, которые объясняют рейтинги.

В системах коллаборативной фильтрации полезность $u(i; j)$ элемента j для пользователя i оценивается на основе полезностей $u(i'; j)$, назначенных элементу j этими пользователями $i' \in \mathcal{S}$, которые «похожи» на пользователя i , где \mathcal{S} представляет набор пользователей [Adomavicius G., etc. 2005].

Когда мы имеем дело с данными, которые состоят из явных предпочтений пользователей, например, оценки, одобрения, лайки и т.п., мы можем использовать Explicit matrix factorization. Для нашего случая этот подход не применим, так как мы не имеем явных оценок, поэтому мы применили Implicit Matrix Factorization. Существует много разных подходов к работе с неявными данными. MLlib реализует особый подход, который отображает пользователей i' и элементы в пространство скрытых факторов в d измерениях, где d представляет размерность пространства. В пространстве скрытых факторов взаимодействие пользователя и элемента моделируются как внутренние рейтинги: матрица предпочтений \mathcal{P} и матрицу весов доверия \mathcal{C} . Более формально, каждый элемент представлен вектором $q_j \in \mathbb{R}^d$, и каждый пользователь представлен вектором $p_i \in \mathbb{R}^d$. Каждый рейтинг прогнозируется произведением векторов, приведенных в уравнении 1.1, и ошибка обучения, измеренная для $(i; j)$ -го рейтинга, который присваивается элементу j пользователем i , определяется в уравнении 1.2 [Koren Y. 2010], [Takacs G, 2008]:

$$\hat{r}_{ij} = q_j^T p_i, \quad (1.1)$$

$$e_{ij} = r_{ij} - \hat{r}_{ij}, \quad (1.2)$$

где \hat{r}_{ij} обозначает прогноз рейтинга i -го пользователя, который бы он

дал j -ому элементу, а e_{ij} обозначает ошибку обучения для этой пары элемент-пользователя.

Ключом к модели является поиск векторов q_j и p_i .

Чтобы узнать фактор-вектора p_i и q_j , модель должна минимизировать регуляризованную ошибку на множестве известных рейтингов \mathfrak{R} , как указано в уравнении 1.3 [Koren Y. 2009]:

$$\min_{q^*, p^*} \sum_{(i,j) \in R} (r_{ij} - q_j^T p_i)^2 + \lambda (\|q_j\|^2 + \|p_i\|^2). \quad (1.3)$$

В уравнении 1.3 R обозначает известный набор рейтингов. Система изучает модель путем подбора ранее заданных рейтингов. Однако, очень важно уметь делать прогнозы для новых товаров, не имеющих рейтинг; так что система должна избегать переоснащения путем регуляризации изученных параметров. Константа λ в этом уравнении контролирует степень регуляризации.

Предположим, что рейтинг товара фактически является количеством покупки данного товара каждым пользователем. Чтобы спроектировать модель, нам необходимо выявить скрытые отношения между пользователями и товарами.

Неявная модель создает две матрицы: матрица \mathcal{P} информирует нас о том, что товар был куплен пользователем, а матрица \mathcal{C} представляет весовой коэффициент достоверности в виде количества покупок (как правило, чем больше пользователей купили данный товар, тем выше достоверность того, что им на самом деле это нравится). Матрица, которую пытается аппроксимировать модель, является матрицей предпочтений \mathcal{P} .

В результате вычисляя точечное произведение вектора пользователя и фактора-элемента будет получена оценка предпочтения пользователя для элемента. Чем ближе эти оценки к нулю, тем выше точность оценки предпочтения пользователя. Высокая достоверность оценки предпочтения пользователя или соответствие между пользовательскими и товарными факторами приводит к рекомендации.

3. Описание набора данных

Чтобы проиллюстрировать описанные выше концепции, мы использовали набор данные, описывающий поведение пользователя на сайте электронной коммерции, в виде просмотров страниц и покупок товаров. Он содержит набор сырых данных, в котором каждая запись представляет собой строку, разделенную через запятую, представляющую событие на сайте электронной коммерции:

– Timestamp – метка времени. Часть метки времени находится в формате эпохи Unix (timestamp), например, 1548835990 будет преобразован в среду, 30 января 2019 8:13:10 GMT.

– Visitorid (идентификатор посетителя) - уникальный пользователь, просматривающий веб-сайт.

– Event – действие пользователя, которое может включать в себя: просмотр товара, добавление товара в корзину, покупку.

– Itemid – уникальный идентификатор товара для конкретного сайта электронной коммерции.

– Transactionid - идентификатор транзакции, который будет иметь значение, только если пользователь совершил покупку: 1433193915008, 552148, transaction, 81345,5444.

Набор данных для обработки в рекомендательной системе взят с общедоступной платформы данных Kaggle Datasets. Данные были собраны с реального веб-сайта электронной коммерции. Это необработанные данные, то есть без каких-либо преобразований содержимого. Для обеспечения конфиденциальности, все значения представлены в хешированном виде.

Набор данных содержит данные о поведении пользователей, т. е. такие события, как клики, добавление в корзину, транзакции, которые были собраны в течение 4,5 месяцев 2015 года. Например:

– «143969400000,1,view,100» означает, что пользователь с visitorId = 1, щелкнул товар с id = 100 при 143969400000 (метка времени Unix).

– «143969400000,2, transaction,1000,234» означает, что пользователь с visitorId = 2 приобрел товар с id = 1000 в транзакции с id = 234 при 143969400000.

Посетитель может совершить три типа событий, а именно: просмотр товара, добавление в корзину или покупку товара. В общей сложности набор данных включает в себя: 1 048 575 событий, включая 1 013 165 просмотров товаров, 26 756 добавлений в корзину и 8 654 покупки, произведенных 556 528 уникальных посетителей сайта электронной коммерции предполагая, что нет одинаковых пользователей с разными идентификаторами посетителей. Набор данных содержит 157 270 уникальных товаров.

Для имитации обработки потоков данных в режиме реального времени мы использовали библиотеку Spark Streaming, которая отвечает за получение данных из входного источника и преобразование необработанных данных в дискретизированный поток discretized stream (DStream), состоящий из Spark RDD. Затем мы создали пользовательское приложение Spark Streaming, которое будет обрабатывать этот поток

событий. Сначала мы создали StreamingContext и сокетный поток, а затем применили map-трансформацию, чтобы разбить необработанный текст и создать кортеж (пользователь, продукт, событие). Далее мы использовали foreachRDD для применения произвольной обработки к каждому RDD в потоке, чтобы вычислить желаемые метрики и вывести их на консоль.

4. Обучение модели с использованием неявных данных

Чтобы сократить число переменных, по которым будет проводиться машинное обучение, создадим низкоразмерное представление матрицы рейтинга. То есть, чтобы получить q и p , матрицу S мы уменьшили до размера k , используя разложение по сингулярным значениям (SVD) матрицы M : $m \times n$ (действительное или комплексное) является факторизацией вида $U \Sigma V^*$, где U - матрица $m \times R$. Σ - прямоугольная диагональная матрица $R \times R$ с неотрицательными действительными числами на диагонали, а V - единичная матрица $n \times r$. r равен рангу матрицы M .

После трансформации необработанных данных было проведено обучение модели. В нашем эксперименте для обучения модели набор данных был разбит на две части: обучающий набор данных 80%, и корректирующий набор данных 20%. Обучающие данные используются в алгоритме для изучения характеристик набора данных, а данные проверки и тестирования полностью не видны для алгоритма. Затем набор тестирующих данных используется для внесения улучшений и используется в качестве буфера, но здесь данные испытаний все еще не видны.

Разделение набора данных на обучающий набор данных и тестирующий набор данных с соотношением 80:20, показан в следующих строках кода:

```
val Array(training, test) = ratings.randomSplit(Array(0.8, 0.2))
training.cache()
test.cache()
```

На выходе RDD, мы имели следующую схему: userID – Int, itemID – Int, rating – Double. Рейтинг товара вычисляется на основе историй взаимодействия пользователя с данным товаром.

Стандартный подход Matrix Factorization в MLlib имеет дело с явными рейтингами. Для работы с неявными данными мы использовали метод trainImplicit со следующими параметрами:

```
val rank = 10 // количество скрытых факторов в модели ALS
val numIterations = 10 // количество итераций алгоритма ALS
```

```
val alpha = 0.01 //базовая достоверность в наблюдениях за предпочтениями
```

```
val lambda = 0.01 //параметр регуляризации
```

```
val model = ALS.trainImplicit(ratings, rank, numIterations, lambda, alpha)
```

Параметр `rank` определяет число скрытых признаков в матрицах аппроксимации низкого ранга. Как правило, чем больше число факторов, тем лучше, но для большого количества пользователей или элементов он напрямую повлияет на использование памяти вычислительной системы и количество необходимых данных для обучения. Поэтому в нашей задаче это было компромиссное решение.

Обычно сходимость модели ALS к достаточно хорошему решению происходит уже после относительно небольшого количества итераций. И хотя каждая итерация в ALS гарантированно уменьшает ошибку восстановления матрицы оценок, параметр `numIterations` равный 10 является хорошим значением по умолчанию.

Также был установлен дополнительный параметр `alpha`, контролирующий базовый уровень достоверности, применяя взвешивание. Более высокий уровень альфа делает модель более уверенной в том факте, что отсутствующие данные равносильны отсутствию предпочтений для соответствующей пары пользователь-элемент.

Значение параметра регуляризации `lambda` было выбрано с помощью методов тестирования и перекрестной проверки.

Обучая такую модель, мы пытались найти наилучшее ранговое приближение d к наблюдаемой целевой матрице R размера $N \times M$, где d – это число скрытого векторного измерения, N и M – соответственно количество пользователей и элементов в системе.

В результате работы модели рекомендация принимает форму списка топ- k , то есть набора k -элементов, имеющие с наибольшую вероятность того, что они понравятся пользователю. Это делается путем вычисления прогнозируемой оценки для каждого элемента и ранжирования списка на основе этой оценки.

5. Оценка работы рекомендательной системы

Для проверки работы рекомендательной системы использовались следующие метрики точности: MSE (Mean Square Error) – среднеквадратичная ошибка; RMSE (Root Mean Square Error) – среднеквадратическое отклонение.

Эти метрики позволяют оценить разницу между реальным рейтингом и рейтингом, предсказанным рекомендательной системой. Вычисление

ошибок происходило на наборе данных из тестовой выборки.

Определим вектор предсказаний как \hat{Y} , а вектор наблюдаемых значений предсказываемой переменной Y , тогда метрика MSE вычисляется по формуле 4.1 [Dua R., etc. 2017].

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2. \quad (4.1)$$

Значение MSE стремится к нулю. Если $MSE = 0$, то система предсказывает параметр с идеальной точностью, что, как правило, невозможно.

RMSE вычисляется выделением квадратного корня из среднего квадрата различий между предсказанием и фактическим значением по формуле 4.2 [Dua R., etc. 2017]:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2}. \quad (4.2)$$

Данная метрика является одной из популярных метрик, так как ошибки, выявленные этой метрикой, могут оказать значительное влияние на решение пользователя. Значение RMSE всегда неотрицательно и чем ближе к нулю, тем идеальнее соответствует данным. Предпочтительнее низкие значения RMSE.

В результате перекрёстной проверки и выбора оптимальных гиперпараметров модели для алгоритма коллаборативной фильтрации были получены следующие значения метрик точности: $MSE = 7.89 \cdot 10^{-4}$, $RMSE = 0.028$.

Заключение

В рамках данной работы был произведен анализ существующих алгоритмов работы с неструктурированными данными. Исследованы алгоритмы коллаборативной фильтрации наборов данных с неявной обратной связью, что является очень распространенной ситуацией, но недостаточно исследованной и реализованной. Алгоритм был протестирован как часть крупномасштабной системы рекомендаций основанной на кластере Spark, интегрированного с Hadoop. Реализация имитационной модели машинного обучения с использованием неявных данных показала, что пользователи склонны выбирать товары, которые в скрытом векторном пространстве находятся рядом друг с другом. Учитывая достоинства и недостатки различных подходов к рекомендательным системам, для создания рекомендательной системы электронной коммерции лучше подходят алгоритмы коллаборативной фильтрации, основанные на модели. Данный подход может эффективно обрабатывать масштабируемость и разреженность данных, что является важным критерием при работе с большим набором разреженных данных

(более миллиона записей о действиях пользователя). При этом, алгоритмы коллаборативной фильтрации, основанные на модели дает более точные рекомендации, чем подход, основанный на памяти.

Малые значения параметров MSE и RMSE, говорят о высокой точности прогноза и о том, что данный алгоритм достаточно эффективно можно использовать для низкоразмерной матрицы, он позволяет снижать размерность признакового пространства и может лучше масштабировать по сравнению с другими подходами, благодаря гибкости в решении реальных задач.

Список литературы

[**Adomavicius G., et al. 2005**] Adomavicius G. and Tuzhilin A.. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE transactions on knowledge and data engineering, pages 734–749, 2005.

[**Das A., et al.2007**] Das A., Datar M., Garg A., Rajaram S. Google news personalization: Scalable online collaborative filtering. In Proc. of WWW'07: the 16th International Conference on World Wide Web, Banff, Alberta, Canada, pages 271–280, 2007.

[**Koren Y. 2009**] Koren Y., Bell R., and Volinsky C.. Matrix factorization techniques for recommender systems. IEEE Computer, 42(8):30–37, 2009.

[**Koren Y. 2010**] Koren Y. Collaborative filtering with temporal dynamics. Communications of the ACM, 53(4):89–97, 2010.

[**Takacs G, 2008**] Pilszky I., Nemeth B., and Tikk D. Matrix factorization and neighbor based algorithms for the netflix prize problem. In Proceedings of the 2008 ACM conference on Recommender systems, pages 267–274. ACM, 2008.

УДК 007.5:519.816:681.3.016

АЛГОРИТМ ПОСТРОЕНИЯ ДЕРЕВЬЕВ РЕШЕНИЙ ПРИ НАЛИЧИИ ПРОТИВОРЕЧИЙ В ДАННЫХ¹

В.Н. Вагин (*vagin@appmat.ru*),

А.В. Крупецков (*erne100-che-guevara@yandex.ru*),

М.В. Фомина (*m_fomina2000@mail.ru*)

Национальный Исследовательский Университет «МЭИ»,

Москва

В работе рассматривается проблема использования алгоритмов машинного обучения в задачах индуктивного формирования понятий при работе с противоречивыми данными. Рассматривается алгоритм построения нечетких деревьев решений UD3, показаны его преимущества при обработке противоречий в данных по сравнению с известными алгоритмами построения деревьев решений. Вводятся правила принятия решений в случае неоднозначной классификации примеров. Даны результаты машинных экспериментов.

Ключевые слова: обобщение, деревья решений, противоречия в данных, бинарные векторы, нечеткие продукционные правила.

Введение

Одной из важных проблем в современных интеллектуальных системах обработки данных, системах принятия решений и системах классификации является проблема индуктивного формирования понятий, или обобщения. Обобщение связано с поиском логических закономерностей на основе построения правил, способных объяснить имеющиеся факты и предсказывать новые. Таким образом, целью обобщения является построение совокупности таких логических связей - классификационной модели. Когда классификационная модель построена, на основании найденных закономерностей мы можем

¹ Работа выполнена при финансовой поддержке РФФИ, гранты № 17-07-00442а, 18-01-00201, 18-29-03088-мк

отнести новые, ранее не рассмотренные объекты к какому-либо классу [Вагин и др., 2008].

1. Постановка задачи индуктивного формирования понятий

Формулировка задачи обобщения понятий по признакам дана в [Вагин и др., 2008]. Пусть в интеллектуальной системе S есть $O = \{o_1, o_2, \dots, o_R\}$ – множество объектов, каждый из которых задан вектором признаков (атрибутов). Количество атрибутов фиксировано, значения атрибутов могут быть числовыми, логическими, символьными.

Среди множества O всех объектов, представленных в некоторой системе S , выделим V - множество объектов, относящихся к понятию (классу) и W - множество объектов, не относящихся к понятию. Будем рассматривать случай, когда $O = V \cup W$, $V \cap W = \emptyset$. Построим обучающую выборку $K = K^+ \cup K^-$, где $K^+ \subset V$ и $K^- \subset W$. На основании обучающей выборки K надо построить правило, разделяющее положительные и отрицательные объекты обучающей выборки. Решающее правило является корректным, если оно в дальнейшем успешно распознаёт объекты, не вошедшие первоначально в обучающую выборку.

Известен ряд алгоритмов, формирующих решающие правила в виде дерева решений, или набора продукционных правил. Прежде всего, это алгоритмы ID3 и C 4.5 [Quinlan, 1986], [Quinlan, 1996], алгоритм CN2 [Clark et al., 1989] и ряд других. Одной из проблем для этих алгоритмов являются сложности, возникающие при обработке как неполной, так и противоречивой информации.

2. Возможности различных алгоритмов обобщения по обработке зашумленных данных

Результатом работы алгоритмов, названных выше, являются обобщенные понятия, представленные в виде деревьев решений либо набора продукционных правил. Решающее дерево (Decision tree) - это дерево, которое каждому входному примеру ставит в соответствие выходное значение, при этом строится путь от корневой вершины к одной из конечных вершин. В каждой из промежуточных вершин (узлов) выполняется проверка условий, а конечные вершины (листья) взвешены именами решения (обычно это имя класса, к которому отнесён пример).

Необходимо, чтобы в каждой промежуточной вершине результаты проверки условий были взаимоисключающими и исчерпывающими.

Выбор последовательности проверок условий при движении от корня

к листьям дерева в работах [Quinlan, 1986], [Quinlan, 1996] определяется критериями, которые связаны с понятием энтропии. Одной из проблем для вышеназванных алгоритмов являются сложности, возникающие при обработке неполной и противоречивой информации. В работах [Vagin et al., 2008], [Vagin et al., 2011] были детально исследованы вопросы влияния шума на эффективность классификационных моделей, в [Антипов и др., 2014] показано, что наиболее существенное влияние на снижение классификационной точности оказывает наличие в обучающей выборке, на основе которой строится дерево решений, противоречивых примеров, то есть примеров, приписанных разным классам, при полном совпадении информативных атрибутов.

Возможность обработки противоречивых примеров в данных может быть реализована с помощью алгоритмов, основанных на теории приближенных множеств (*tough sets*) [Pawlak, 2002]. При этом классификационная модель, представленная наборами продукционных правил, дает как точную классификацию (пример *принадлежит/ не принадлежит* классу), так и приближенную (пример *возможно принадлежит* классу). В последнем случае не дается числовых оценок уверенности в принятом решении.

Далее мы рассмотрим новый алгоритм построения классификационной модели в виде дерева решений, который стремится преодолеть вышеназванные проблемы.

3. Описание алгоритма, работающего с противоречиями

В основу метода был положен алгоритм UD3 (Uncertain Decision Tree), разработанный С.Фахрамадом и С.Джафари [Fakhrmahad, Jafari, 2008]. Этот алгоритм основан на идеях алгоритма ID3, однако, построение дерева решений и последующая классификация тестовых примеров связаны с использованием дополнительной методики, основанной на сравнении битовых строк, сопоставленных атрибутам обучающей выборки. Основная цель здесь – обеспечить правильное решение при классификации примеров в случае, если набор данных содержит противоречивые примеры.

Пусть атрибут A принимает дискретное значение a из области допустимых значений $Dom(A)$. Для обучающей выборки K построим двоичный вектор, который является бинарной строкой, имеющей длину N , $N=|K|$ (этот вектор обозначается как $VCode(A=a)$). Каждый бит в этом векторе связан с конкретным примером и устанавливается в 1, если для этого конкретного примера атрибут A имеет значение a и в 0, в противном случае. Таким образом, значения $\{1, 0\}$ в векторе $VCode$ обозначают

истину или ложь логического выражения ($A=a$) на всех примерах. Так же можно получить векторы BCodes для сложных логических выражений, применяя операцию & (логическое И) между каждым членом выражения. Например,

$$\begin{aligned} & \text{BCode}(A_1 = a_1, A_2 = a_2, \dots, A_n = a_n) = \\ & = \text{BCode}(A_1 = a_1) \& \text{BCode}(A_2 = a_2) \& \dots \& \text{BCode}(A_n = a_n) \end{aligned}$$

Очевидно, аналогичные бинарные векторы можно построить для атрибута, на основе которого определяется класс каждого примера в обучающей выборке. Если этот атрибут бинарный, будут построены два бинарных вектора, обозначим их далее, как V^+ и V^- . В случае, если обучающая выборка не содержит противоречий, V^- будет инверсией V^+ .

В основу алгоритма UD3 положена идея алгоритма [Quinlan, 1986], исходными данными является обучающая выборка K , содержащая N примеров, результатом работы алгоритма будет дерево решений. Построение дерева решений выполняется на основе следующей процедуры рекурсивного характера:

1. Выбирается атрибут для корневого узла дерева, и формируются ветви для каждого из возможных значений этого атрибута. С каждой ветвью дерева связывается двоичный вектор BCode.

2. Дерево используется для классификации примеров обучающего множества. Если все примеры на некотором листе принадлежат одному классу, то этот лист помечается именем этого класса.

3. Если все листья помечены именами классов, алгоритм заканчивает работу. В противном случае узел помечается именем очередного атрибута, и создаются ветви для каждого из возможных значений этого атрибута, с каждой новой ветвью связывается двоичный вектор BCode.

Переход к шагу 2 алгоритма.

Критерием для выбора очередной проверки в узле дерева решения, начиная с корневого узла, является поиск атрибута, наиболее тесно связанного с решающим атрибутом, определяющим класс каждого примера. Рассмотрим этот критерий.

Пусть атрибут A_i принимает значения из области $\text{Dom}(A_i)$. Обучающее множество может быть разделено на подмножества, в соответствии с количеством значений атрибута A_i . В случае, если A_i – бинарный атрибут, получим разбиение K на 2 подмножества: $K = K^1 \cup K^2$. Для каждого из двух подмножеств ищем n^+ – количество примеров, принадлежащих искомому классу и n^- – количество примеров, не принадлежащих искомому классу. Критерием близости данного атрибута A_i и решающего атрибута будет отношение $q_i = (c_i^+ +$

$c_2)/N$, где $c_1 = \max(n1^+, n1^-)$, $c_2 = \max(n2^+, n2^-)$, N - общее число примеров в обучающей выборке.

Атрибут A_i , обладающий наилучшим значением q_i , приписывается очередному узлу дерева решений. Формируются ветви дерева, в зависимости от количества значений атрибута, с каждой ветвью дерева связывается вектор $VCode$, введенный выше. Величины n^+ и n^- определяются как количество единиц в векторах $VCode$.

4. Обработка противоречий в данных

Наличие противоречивых примеров в обучающей выборке K может привести к тому, что в ходе построения дерева решений алгоритмом UD3 все информативные атрибуты уже использованы для проверок, но при этом невозможно присвоить листу имя определённого класса. Такое дерево решений может содержать правила, дающие неоднозначную классификацию, и будет называться нечетким. Таким образом необходим дополнительный метод разрешения противоречий на этапе классификации. Для этого используются бинарные векторы $VCodes$, связанные с ветвями нечеткого дерева решений.

Пусть в построенном дереве решений существует лист L с неоднозначной «классовой принадлежностью». Рассмотрим путь от корня дерева к этому листу и построим конъюнкцию векторов $VCode$, приписанных дугам в этом пути. Полученный вектор $Vant$ показывает примеры, чьи атрибуты полностью совпадают со значениями атрибутов, существующих в пути от корня к L . Правило, сопоставленное такому пути, даёт неоднозначную классификацию, поэтому критерием принятия решения при классификации будет наименьшее число противоречий.

Используя векторы $VCodes$ найдем примеры, нарушающие правило, если мы выделим (+) как целевой класс. Количество таких примеров (обозначим его как e^+) определяется количеством единиц в $VCode$.

Аналогично найдем примеры, нарушающие правило, если мы выделим (-) как целевой класс. Величина e^- дает количество примеров, нарушающих правило, если целевым классом выбран класс (-). Как результат, узлу L будет приписано имя класса с наименьшим числом противоречий.

Рассмотрим правило $rule: X \rightarrow Y$. Здесь X представляет собой конъюнкцию простых условий, определяющих путь от корня к листу на дереве решений. Y – решающий атрибут, приписанный конечной вершине дерева решений; при наличии противоречий в обучающей выборке конечной вершине могут быть приписаны имена сразу двух классов. Для оценки правила $X \rightarrow Y$ мы используем две хорошо известных метрики –

Confidence и *Support*, которые являются основными мерами в извлечении часто встречающихся примеров и ассоциативных правил. *Support* для правила $X \rightarrow Y$ определяется как доля примеров в наборе данных, удовлетворяющих X и Y , одновременно. *Confidence* для правила $X \rightarrow Y$ определяется как $Support(X \cap Y)/Support(X)$, например доля всех примеров удовлетворяющих X , которые так же удовлетворяют и Y . Используем эти две величины для определения фактора уверенности (*CF*) для составленных правил:

$$CF(\text{rule}) = Confidence(\text{rule}) * Support(\text{rule}).$$

Если дерево решений включает в себя множество длинных путей от корня до листовых узлов, т. е. X содержит большое число конъюнкций, может быть предпочтительно иметь более короткие правила, даже если они имеют меньшие значения (*CF*). Для этого мы можем ввести ограничения на количество сомножителей в части X продукционного правила. Используя фактор (*CF*) можно подобрать подходящие значения для такого ограничения, когда точность классификации тестовых примеров на этапе экзамена снижается на доли процента.

5. Результаты машинного эксперимента

Исследование способности алгоритма UD3 работать с противоречивой информацией проводилось в системе формирования обобщенных понятий "Induction of Decision Tree with restoring Unknown Values" (IDTUV3), описанной в [Fomina et al., 2014]. IDTUV3 включает процедуры обращения к различным алгоритмам формирования обобщенных понятий. Был подключен отдельный блок, реализующий алгоритм UD3. Для исследования влияния шума на формирование обобщенных правил в программном комплексе используется блок внесения шума в данные.

В работе представлены результаты эксперимента, выполненных над наборами данных из известных коллекций множества тестовых данных из California University of Informatics and Computer Engineering "UCI Machine Learning Repository" [UCI, 1998].

Перечислим основные этапы эксперимента.

1. Выбор и загрузка обучающего множества из наборов данных, хранящихся в БД.

2. Выбор уровня шума, внесенного в обучающие и тестовые множества.

3. Построение классификационной модели (дерево решений или набор продукционных правил) на основе обучающих выборок.

4. Использование построенной классификационной модели для тестирования примеров, входящих в экзаменационные выборки. Сбор

статистики и определение точности классификации экзаменационных данных.

Машинный эксперимент должен был оценить способности разработанных алгоритмов строить обобщенные понятия при наличии противоречий в данных. Поэтому был предусмотрен следующий вариант внесения шума в данные. Шум вносился равномерно во всю таблицу, использовалась модель шума «искажение значений». В случае, если искажению подвергался решающий атрибут, в обучающую выборку добавлялась аналогичная строка с противоречивым значением в решающем атрибуте.

На рисунке 1 для набора данных «Monks-1»: представлена зависимость точности классификации тестовых примеров от уровня шума, внесенного в обучающую выборку. Сравниваются два алгоритма C 4.5 и UD3.

На рисунках 2 и 3 приведены результаты тестирования по выборкам Monks-2 и Monks-3.

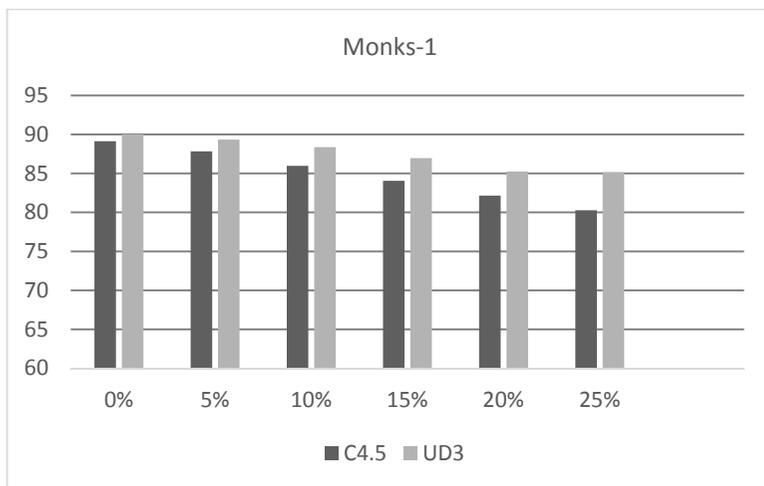


Рис. 1. Зависимость точности классификации тестовых примеров (%) от шума в обучающей выборке Monks-1

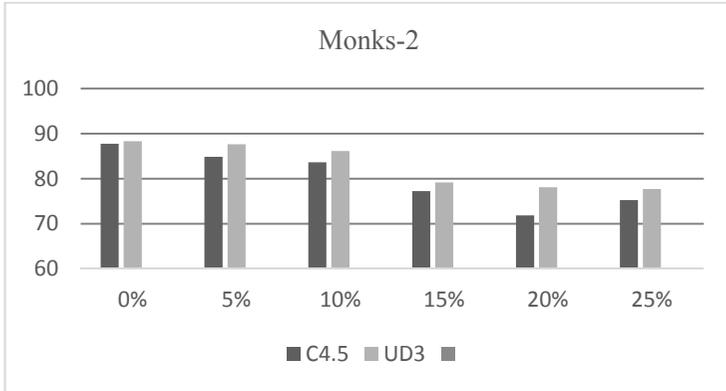


Рис. 2 Зависимость точности классификации тестовых примеров (%) от шума в обучающей выборке Monks-2

На каждой диаграмме можно видеть, как с ростом числа противоречивых примеров в обучающем множестве снижается точность классификации примеров экзаменационных выборок. При этом алгоритм UD3 имеет преимущество перед алгоритмом C 4.5 особенно при больших уровнях шума. Таким образом, даже с появлением в обучающей выборке большого количества противоречивых примеров при высоких (свыше 10%) уровнях шума алгоритм UD3 имеет преимущество.

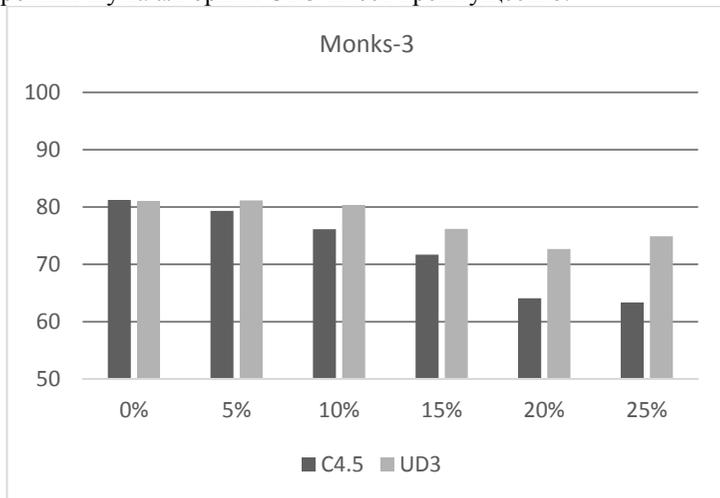


Рис. 3 Зависимость точности классификации тестовых примеров (%) от шума в обучающей выборке Monks-3.

Пусть в результате работы алгоритма UD3 получено нечеткое продукционное правило, связанное с листом L. Листу L приписаны e примеров нескольких классов и e_{min} - наименьшее число примеров одного класса среди примеров e . Тогда значение $p = e_{min} / e$ дает отношение количества примеров наиболее слабого класса к общему числу примеров, приписанных одному листу. В [Fakhrahmad, Jafari, 2008] предлагается использовать пороговое значение p^* , и при условии $p < p^*$ продукционное правило можно рассматривать как полностью точное. Экспериментально в [Fakhrahmad, Jafari, 2008] было выбрано наиболее удачное значение $p^*=0.02$ для такого преобразования.

Перечислим наиболее важные особенности алгоритма UD3. Использование для вычислений битовых строк, BCodes, позволяет уменьшить вычислительную нагрузку благодаря замене операций логарифмирования, применяемых в аналогах на логические операции.

В алгоритме также предусмотрены механизмы для предотвращения излишнего разрастания дерева и, как следствие переобучения, что позволяет уменьшить число вычислений при незначительном уменьшении точности. Это делает данный алгоритм перспективным для дальнейшего развития.

Заключение

В работе рассматривалась задача обобщения информации и исследовалось влияние шума в исходных данных на эффективность классификационных моделей, полученных с помощью алгоритмов обобщения. Рассмотрен наиболее сложный вариант шума в таблицах баз данных, связанный с наличием противоречий в обучающей выборке.

Были исследованы возможности использования алгоритма построения нечетких деревьев решений UD3 для снижения влияния шума такого типа. Показаны преимущества данного алгоритма по сравнению с алгоритмами Куинлана ID3 и C4.5. В дальнейшем будет рассмотрено влияние на работу алгоритма обобщения ограничений на количество условий в левой части продукционных правил. Также необходимо провести сравнение возможностей алгоритма UD3 с алгоритмами, основанными на теории приближенных множеств.

Список литературы

[Антипов и др., 2014] Антипов С.Г., Старостина Л.А., Фомина М.В. Проблема формирования обобщенных понятий при наличии шума в решающих атрибутах. // Четырнадцатая национальная конференция по искусственному интеллекту с международным участием КИИ 2014 (24-27 сентября 2014 г., г. Казань, Россия): Труды конференции., 2014. Т.2

[Вагин и др., 2008] Вагин В.Н. Головина Е.Ю., Загорянская А.А., Фомина М.В. Достоверный и правдоподобный вывод в интеллектуальных системах. /Под ред. В.Н. Вагина, Д.А. Пospelова. – Изд. 2-е, М.: ФизМатЛит, 2008.

[Clark et al., 1989] P.Clark, T.Niblett. The CN2 Induction Algorithm, 1989

[Fakhrahmad, Jafari, 2008] Fakhrahmad S.M., Jafari S. Uncertain Decision Tree Inductive Inference, World Congress of Engineering, 2008

[Fomina et al., 2014] Fomina M., Morosin O., Vagin V. Argumentation Approach and Learning Methods in Intelligent Decision Support Systems in the Presence of Inconsistent Data/ ICCS 2014. 14th International Conference on Computational Science. Procedia computer science. Vol. 29. Elsevier, 2014

[Pawlak, 2002] Pawlak Z. Rough sets and intelligent data analysis / Information Sciences, Elsevier Science, Nov. 2002, vol. 147, iss. 1

[Quinlan, 1986] Quinlan J.R. Induction of Decision Trees // Machine Learning. 1986. № 1(1).

[Quinlan, 1996] Quinlan J.R. Improved Use of Continuous Attributes in C 4.5. // Journal of Artificial Intelligence Research. 1996. №4.

[UCI, 1998] UCI Repository of Machine Learning Datasets. <http://archive.ics.uci.edu/ml/>

[Vagin et al., 2008] Vagin V.N., Fomina M.V., Kulikov A.V.: The Problem of Object Recognition in the Presence of Noise in Original Data. In: 10th Scandinavian Conference on Artificial Intelligence SCAI 2008. pp. 60-67(2008).

[Vagin et al., 2011] V. Vagin and M. Fomina, Problem of Knowledge Discovery in Noisy Databases // International Journal of Machine Learning and Cybernetics. 2011. № 3(2).

ИССЛЕДОВАНИЕ ПРОТИВОРЕЧИЙ В ЕСТЕСТВЕННЫХ РАССУЖДЕНИЯХ НА ПРИМЕРАХ МЕТАФОР И ПРЕСУППОЗИЦИЙ¹

Б.А. Кулик (*ba-kulik@yandex.ru*)
Институт проблем машиноведения РАН,
Санкт-Петербург

В докладе сравниваются противоречия в естественных рассуждениях с формальным противоречием. Анализируются противоречия в логических моделях метафор и пресуппозиций. Устанавливается, что модель пресуппозиции тоже содержит противоречие, но, в отличие от метафоры, здесь его можно элиминировать без потери смысла выражения. Рассматривается связь пресуппозиции с аномалией противоречия в базах знаний.

Ключевые слова: метафора, пресуппозиция, логическая модель, противоречие, аномалия противоречия

Введение

В науке противоречия считаются индикаторами некорректности, и их обнаружение стимулирует новые исследования с целью уточнения и развития наших знаний. Однако можно найти ситуации, когда противоречия оцениваются как достоинства некоторых текстов. К ним относится метафора, относительно которой будет показано, что она невозможна без скрытого противоречия. Кроме того, будут приведены доводы в пользу того, что противоречие также содержится и в логической модели пресуппозиции, но, в отличие от метафоры, его можно элиминировать без искажения смысла текста.

Рассмотрим особенности противоречий в естественных рассуждениях. В математической логике противоречие распознается как ситуация, при которой логическая формула, представляющая рассуждение, является тождественно ложной [Mendelson, 2015]. В естественных рассуждениях противоречием считается ситуация, когда некий предположительно

¹ Работа выполнена при финансовой поддержке РФФИ (проект № 18-07-00132 и 19-08-00079).

существующий объект обладает несовместимыми свойствами. Например, два предложения «Эти грибы ядовиты» и «Эти грибы не ядовиты» совместно считаются противоречивыми, хотя при формальном подходе к их анализу противоречия нет. Для логического анализа этих и других предложений в данном тексте будем использовать сравнительно простую систему из [Кулик, 2001]. Применительно к формулам исчисления высказываний ее можно сформулировать в виде следующих правил:

1. Высказывания типа « A есть B » и « A есть не B » выражаются формулами $A \supset B$ и $A \supset \neg B$;

2. Используются два правила вывода: *контрапозиции* и *транзитивности*;

3. Если из посылок выводима формула типа $A \supset \neg A$, то она указывает на то, что в рассуждении содержится *коллизия парадокса*.

Система вывода в [Кулик, 2001] значительно шире и включает в себя не только перечисленные выше структуры и правила, но для анализа примеров, используемых в тексте, приведенных правил достаточно.

Проанализируем пример. Обозначим A – эти грибы, B – ядовитые. Тогда данные предложения выражаются формулами $A \supset B$ и $A \supset \neg B$. По правилу контрапозиции из $A \supset B$ выводится $\neg B \supset \neg A$, по правилу транзитивности из $A \supset \neg B$ и $\neg B \supset \neg A$ выводится $A \supset \neg A$.

С точки зрения формальной логики полученный парадокс не является противоречием: формула $A \supset \neg A$ не относится к тождественно ложным. В то же время анализ показывает, что логическая переменная «эти грибы» (A) в рассуждении принимает только ложное значение. Парадокс возникает лишь в тех случаях, когда неявно предполагается истинность A . Если это неявное предположение добавить в состав посылок, то получим тождественно ложную формулу $A \wedge (A \supset B) \wedge (A \supset \neg B)$.

Таким образом, в естественных рассуждениях противоречие распознается в случае, когда из посылок выводится отрицание некоторой сущности, которая предполагается истинной. Если эту сущность выразить в виде логической формулы и добавить в состав посылок, то получится формальное противоречие.

1. Логическая модель метафоры

Метафора (при формальном подходе к определению, т.е. без учета ее эстетических характеристик) – это слово (в общем случае – выражение), которое намеренно используется в тексте вместо другого (замещаемого) слова (выражения) на основании некоторого *неполного совпадения* значений этих слов (выражений).

Неполное совпадение значений в определении метафоры существенно,

иначе трудно отличить метафору от синонима. Заметим, что использование метафор в любой науке, скорее, правило, чем исключение (например, эффект сплетен в химических реакциях, черная дыра, солнечная корона, компьютерный вирус, решетки в математике и т.д.).

Попытка сформулировать логическую модель метафоры содержится в [Лагута, 2003]. Здесь метафора определяется как некоторая логическая аномалия и представляет собой свернутое умозаключение (энтимему), т.е. умозаключение с пропущенной посылкой. В качестве примера используется метафора «Адмиралтейская игла» (в цитируемом тексте «игла Адмиралтейства») из поэмы Пушкина «Медный всадник». Очевидно, что «игла» в данном случае замещает слово «шпиль». Предлагается следующее умозаключение.

Меньшая посылка: этот шпиль (S) – очень длинный по отношению к собственному диаметру, прямой, с острым концом (архитектурный элемент) (M).

Большая посылка: некоторые длинные по отношению к собственному диаметру, прямые, с острым концом (орудия) (M) – иглы (P).

Заключение: Шпиль (S) – игла (P).

Заметим, что в приведенном тексте логическая аномалия проявляется не только как энтимема. По правилам силлогистики и формальной логики заключение «Шпиль – игла» нельзя вывести из исходных посылок. В частности, некорректность в том, что большая посылка является частным суждением (слово «некоторые»), поэтому силлогизм оказывается неправильным, а приведенное заключение – не выводимым.

Придется согласиться с тем, что метафора алогична. Рассмотрим тот же пример с другой точки зрения. К введенным выше обозначениям S , M , и P добавим A – архитектурный элемент и I – орудие труда. Соотношения между этими сущностями формулируются в виде следующих посылок:

$S \supset M$; $P \supset M$; $S \supset A$; $P \supset I$; $A \supset \neg I$.

В последней посылке утверждается, что свойства «архитектурный элемент» и «орудие труда» несовместимы.

Для анализа этих посылок воспользуемся системой вывода, изложенной во Введении. Построим некоторые следствия. Из суждения $P \supset I$ по правилу контрапозиции следует $\neg I \supset \neg P$, а из суждений $S \supset A$, $A \supset \neg I$ и $\neg I \supset \neg P$ по правилу транзитивности следует $S \supset \neg P$ (шпиль – не игла). Но если добавить в систему посылок суждение $S \supset P$ (т.е. суждение «шпиль – игла», определяющее метафору), то получим парадокс в виде следствия $S \supset \neg S$.

Эту систему посылок можно выразить как формулу исчисления высказываний: $(S \supset M) \wedge (P \supset M) \wedge (S \supset A) \wedge (P \supset I) \wedge (A \supset \neg I)$.

При замене слова «шпиль» на слово «игла» неявно предполагается, что логическая переменная S (шпиль) истинная. Если эту предпосылку присоединить к вышеприведенной формуле, то получим формально противоречивую формулу:

$$S \wedge (S \supset M) \wedge (P \supset M) \wedge (S \supset A) \wedge (P \supset I) \wedge (A \supset \neg I) \wedge (S \supset P). \quad (1)$$

Спрашивается, какую роль играет противоречие в метафоре? Частично ответ на этот вопрос можно найти в книге [Ricoeur, 1975], в которой обосновывается, что противоречие в метафорах создает напряжение (tension) между терминами, которое и составляет суть метафорического смысла. Очевидно, что это «напряжение» лежит в основе эстетической привлекательности метафоры.

С точки зрения логического анализа подобная ситуация встречается не только в метафорах. Например, в рассуждениях по аналогии разные объекты или сущности отождествляются на основе совпадения некоторых свойств. Подобное отождествление происходит и в некоторых моделях рассуждений по прецеденту [Riesbeck et al., 1989].

С учетом этого имеет смысл обобщить парадокс, возникающий в метафорах, на многочисленные случаи отождествления разных объектов. Назовем его *парадокс подмены*. Пусть имеется некоторый исходный объект O и его аналог A , при этом множество P_C свойств у этих объектов совпадает. Известно также, что объекту O присущи свойства P_O , а объекту A – свойства P_A , при этом данные свойства несовместимы, что можно выразить с помощью формулы $P_A \supset \neg P_O$. Тогда логическую модель подмены можно представить формулой:

$$A \wedge (A \supset P_C) \wedge (O \supset P_C) \wedge (A \supset P_A) \wedge (O \supset P_O) \wedge (P_A \supset \neg P_O) \wedge (A \supset O).$$

В этой формуле подформула $A \supset O$ выражает процедуру отождествления исходного объекта с аналогом, а стоящая слева подформула A – утверждение об истинности аналога. Нетрудно убедиться, что данная формула, также как и подобная ей формула (1), противоречива.

Парадокс подмены не всегда опровергает часто встречающиеся и весьма полезные рассуждения по аналогии или рассуждения по прецеденту. Он появляется лишь в тех случаях, когда обнаруживаются несовместимые свойства отождествляемых сущностей.

2. Логический анализ пресуппозиций

Понятие пресуппозиции (английское presupposition – предположение) весьма часто встречается в научной литературе по логике и философии [Strawson, 1952; Fraassen, 1968; Beaver, 2001], лингвистике [Karttunen et al., 1977; Падучева, 2013], в искусственном интеллекте [Попов, 1982] и т.д.

Пресуппозиция – это утверждение, которое подразумевается (или воспринимается как истинное) при актуализации основного утверждения или вопроса, при этом отрицание (или ложность) основного утверждения не нарушает истинности пресуппозиции. Еще одна особенность пресуппозиции заключается в том, что предположение о ее ложности или несостоятельности влечет потерю смысла основного утверждения.

Например, в предложении *Петр завершил статью* подразумевается, что *Петр до этого работал над статьей*, – это и есть пресуппозиция. Ясно, что отрицание основного утверждения (*Петр не завершил статью*) не влияет на истинность пресуппозиции. Если предположить, что пресуппозиция ложная, то становится ясным, что завершение или не завершение статьи в данной ситуации не имеют смысла.

Пресуппозиции нередко используются в полемике для формулировки «каверзных» вопросов, в которых независимо от ответа подразумевается провинность оппонента. Например, *Ты продолжаешь бить своего отца?* или *Ты собираешься вернуть украденное?*

Далее будем так же, как и в [Beaver, 2001], называть основное утверждение **ассерцией**.

Приводимая во многих современных публикациях логическая модель пресуппозиции была впервые сформулирована в работах [Strawson, 1952; Graassen, 1968]. Так, ван Фрассен рассматривает отношения, которые возникают между пресуппозицией и импликацией. В качестве одного из возможных вариантов он предлагает такую конструкцию:

P – пресуппозиция предложения S , если и только если:

- (a) если S истинно, то P истинно,
- (b) если (не- S) истинно, то P истинно.

Это соотношение можно выразить в виде формулы исчисления высказываний: если P – пресуппозиция предложения S , то для нее справедливо $(S \supset P) \wedge (\neg S \supset P)$.

Однако рассмотрение пресуппозиции в качестве консеквента импликации вызывают сомнения. Во-первых, как отмечено в [Beaver, 2001], можно доказать, что $(S \supset P) \wedge (\neg S \supset P) = P$, т.е. ассерция S оказывается фиктивной логической переменной. Во-вторых, многочисленные примеры пресуппозиций показывают, что по смыслу P является предусловием ассерции, но не наоборот. Как правило, события, выраженные в ассерции, являются продолжением событий, описанных в пресуппозиции, и поэтому их рассмотрение в качестве предусловия (или antecedента) некорректно. Для тех многочисленных видов пресуппозиций, в которых она является предусловием, предложим иной по сравнению с цитируемыми источниками подход к формулировке логической модели пресуппозиции,

в котором антецедентом импликации является пресуппозиция.

Тогда появляется следующая трудность: если выразить пресуппозицию как предусловие, то возникает парадокс. В самом деле, формула $(P \supset S) \wedge (P \supset \neg S)$ равносильна формуле $\neg P$, т.е. выводится безусловная ложность пресуппозиции. Хотя неформальный анализ многих конкретных пресуппозиций показывает, что парадокса на самом деле нет. Типичный пример позволяет найти ключ к решению этой проблемы.

Рассмотрим предложение (*S*) *Антон опоздал в школу*. Пресуппозицией для него является предложение (*P*) *Антон направлялся в школу*. Ясно, что прежде, чем опоздать (либо прийти вовремя), Антон должен был направляться в школу, т.е. *P* здесь в качестве предпосылки вполне правомерно. Однако при этом не учитываются возможные причины опоздания (проспал, встретил друзей и пообщался с ними, помогал перейти дорогу старушке и т.д.). Их можно отобразить только с помощью новой логической переменной. Назовем эту логическую переменную «переключателем ассерции» и сформулируем следующую гипотезу.

Гипотеза 1. Если *P* – пресуппозиция предложения *S*, то существует и может быть найден **переключатель ассерции** *R* со следующими свойствами: выражение $P \wedge R$ является предпосылкой предложения *S*, а выражение $P \wedge \neg R$ – предпосылкой предложения $\neg S$.

Тогда логическую модель пресуппозиции можно выразить в виде формулы $((P \wedge R) \supset S) \wedge ((P \wedge \neg R) \supset \neg S)$.

Анализ показывает, что в данной формуле фиктивных переменных нет. Кроме того, можно легко проверить, что данная формула не инициирует парадокс.

В то же время предложенный подход не объясняет еще одной «странности» пресуппозиции: если пресуппозиция ложная или отрицается, то ассерция и ее отрицание теряют смысл. Предполагается, что объяснение этого феномена выходит за рамки двузначной логики [Beaver, 2001]. Это означает, что в качестве инструмента анализа необходимо использовать неклассическую логику. Однако можно решить эту проблему и в рамках классической логики, для этого достаточно предположить, что некоторые переменные могут иметь более двух значений. Тогда для формулировки логической модели пресуппозиции необходимо вместо логических переменных *P*, *R* и *S* использовать одноименные одноместные предикаты.

Пусть даны допустимые множества констант для предикатов $P = \{0, 1\}$, $R = \{0, 1\}$, $S = \{a, b, c\}$. Для случая, когда имеет место $P(0)$ (ложная пресуппозиция), рассмотрим ассерцию *S*. Потеря ее смысла в данном случае означает, что в предложенной формализации ассерция

принимает третье значение. Покажем это на примере. Если Антон не направлялся в школу ($P(0)$), то он не мог опоздать ($\neg S(a)$), не мог прийти вовремя ($\neg S(b)$), а просто отсутствовал в школе ($S(c)$).

С учетом этого сформулируем следующую логическую модель пресуппозиции.

Если P – пресуппозиция S , то выполняются следующие правила:

(1) если $P(1)$ и $R(1)$, то $S(a)$;

(2) если $P(1)$ и $R(0)$, то $S(b)$;

(3) если $P(0)$, то $S(c)$.

Для этой модели можно сформулировать еще одну гипотезу.

Гипотеза 2. Во всех конкретных случаях, когда отрицание пресуппозиции приводит к «потере смысла» ассерции S , этот смысл находится в виде третьего значения S .

Анализу логических парадоксов посвящен ряд работ Д.А. Бочвара. В частности, в статье [Бочвар, 1944] утверждается, что все парадоксы возникают вследствие добавления к системе аксиом некоторых специальных аксиом, противоречащих аксиомам логики. Однако пример пресуппозиции показывает, что противоречие в ряде случаев возникает не от избытка, а, наоборот, от недостатка некоторых исходных аксиом или предположений. Кроме того, данный пример подтверждает, что в некоторых случаях для анализа противоречий нет необходимости выходить за рамки классической логики.

3. Связь пресуппозиции с аномалией противоречия

Одной из часто встречающихся аномалий баз знаний является аномалия противоречия [Harmelen, 1997]. Рассмотрим обозначения правил, которые в базах знаний (БЗ) выражаются как структуры

$$r_m: B_1 \wedge B_2 \wedge \dots \wedge B_n \rightarrow A,$$

где r_m – имя правила, а B_1, B_2, \dots, B_n, A – атомы.

Каждое правило задано в определенной схеме отношения, а каждая схема отношения характеризуется множеством имен атрибутов. Для правила r_m обозначим $Ant(r_m)$ схему отношения его антецедента, $Cons(r_m)$ – схему отношения его консеквента, а $Val(X_i, r_m)$ – значение атрибута X_i в правиле r_m . Например, атом B_i , относящийся к атрибуту X_i , выражается частью фразы

«Если $X_1 = \dots$ и $X_i = a$ или $X_i = b$ и $X_{i+1} = \dots$ и $X_n = \dots$, то...».

Тогда $Val(X_i, r_m) = \{a, b\}$. Аналогично, консеквент каждого правила можно отобразить как множество значений для определенного атрибута.

Рассмотрим аномалию противоречия. Пусть имеются два правила:

$$r_D: B_1 \wedge B_2 \wedge \dots \wedge B_n \rightarrow D;$$

$$r_F: B_1 \wedge B_2 \wedge \dots \wedge B_n \rightarrow F.$$

При этом $D \cap F = \emptyset$. Тогда правила r_D и r_F иницируют аномалию противоречия. Обычно противоречивые правила корректируются с помощью экспертов, при этом корректировка заключается в том, что одно из этих правил или оба удаляются из БЗ. Однако приведенный выше логический анализ пресуппозиций дает нам возможность корректировать данную аномалию с помощью поиска дополнительных атрибутов, подобных переключателям ассерции.

Обозначим $Ant(*)$ схему отношений антецедентов правил r_D и r_F , а Y – имя атрибута для консеквентов этих правил. Тогда алгоритм поиска переключателя ассерции для правил r_D и r_F будет следующим.

1. Среди множества всех правил БЗ найти множество S правил r_m , у которых $Cons(r_m) = Y$, в этом множестве сформировать подмножества правил S_D и S_F , у которых значением атрибута Y являются соответственно множества $Val(Y, r_D)$ и $Val(Y, r_F)$;

2. Из множеств S_D и S_F исключить правила, для которых не выполняется $Ant(*) \subset Ant(r_m)$ и $Ant(*) \subset Ant(r_n)$ (где \subset – строгое включение, $r_m \in S_D, r_n \in S_F$);

3. Из множеств S_D и S_F исключить правила, у которых значения атрибутов в $Ant(*)$ отличаются от соответствующих значений в противоречивых правилах r_D и r_F ;

4. Из множеств S_D и S_F сформировать множество P пар правил (r_m, r_n) , таких что $r_m \in S_D, r_n \in S_F$, и $Ant(r_m) \cap Ant(r_n) \setminus Ant(*) \neq \emptyset$;

5. В множестве P выполнить для каждой пары (r_m, r_n) и каждого атрибута X_i из множества $(Ant(r_m) \cap Ant(r_n)) \setminus Ant(*)$ следующую проверку $Val(X_i, r_m) \cap Val(X_i, r_n) = \emptyset$;

6. Если в п. 5 ответ “yes”, то для противоречивых правил r_D и r_F атрибут X_i является переключателем ассерции.

Например, в базе знаний могут встретиться следующие 2 правила:

r_D : если бык идет по направлению к субъекту, и субъект показывает быку красную тряпку, то имеется **большой** риск нанесения значительного ущерба здоровью субъекта;

r_F : если бык идет по направлению к субъекту, и субъект показывает быку красную тряпку, то имеется **малый** риск нанесения значительного ущерба здоровью субъекта.

Предположим, что в соответствующей БЗ произведен поиск атрибутов, выполняющих роли переключателей ассерции для этих противоречивых правил. Одним из таких атрибутов может оказаться «местоположение субъекта» (субъект, например, может стоять в чистом поле, либо находиться в кабине бронетранспортера). Другой вариант: атрибут

“профессия субъекта”, если он, допустим, тореадор, то значением атрибута Y является «малый риск», в противном случае – «большой».

Заключение

Рассмотрена логическая модель метафоры, основной особенностью которой является наличие противоречия (парадокса). Чтобы без противоречия объяснить пресуппозицию как предусловие в рамках классической логики, предложено добавить в рассуждение новый фактор, переключатель ассерции. Показана связь пресуппозиции с аномалией противоречия в базах знаний.

Список литературы

- [Beaver, 2001] Beaver D. Presupposition and Assertion in Dynamic Semantics. Stanford: CSLI Publications. 2001.
- [Fraassen, 1968] Fraassen B. van. Presupposition, implication and self-reference // Journal of Philosophy. 1968. V. 65. № 5.
- [Harmelen, 1997] Harmelen F. Applying rule-base anomalies to KADS inference structures // Decision Support Systems. 1997. V. 21, № 4.
- [Karttunen et al., 1977] Karttunen L., Peters S. Requiem for Presupposition // Proceedings of the Third Annual Meeting of Berkeley Linguistic Society. Berkeley, 1977.
- [Mendelson, 2015] Mendelson, E. Introduction to Mathematical Logic. 6th ed. Taylor & Francis Group. 2015.
- [Ricoeur, 1975] Ricoeur P. La métaphore vive. Paris: Éditions du Seuil, 1975.
- [Riesbeck et al., 1989] Riesbeck, C. K. and Schank, R. S. Inside Case-Based Reasoning. Lawrence Erlbaum Assoc., Inc., Hillsdale, NJ. 1989.
- [Strawson, 1952] Strawson P. Introduction to Logical Theory. London. 1952.
- [Бочвар, 1944] Бочвар Д. А. К вопросу о парадоксах математической логики и теории множеств // Матем. сборник, 1944, том 15(57), № 3.
- [Кулик, 2001] Кулик Б. А. Логика естественных рассуждений. – СПб.: Невский диалект. 2001.
- [Лагута, 2003] Лагута О. Н. Метафорология: теоретические аспекты. Ч. 1. Новосиб. гос. ун-т. Новосибирск, 2003.
- [Падучева, 2013] Падучева Е. В. Русское отрицательное предложение. – М.: Языки славянской культуры, 2013.
- [Попов, 1982] Попов Э.В. Общение с ЭВМ на естественном языке. – М.: Наука. 1982.

Список авторов

Абрамов М.В.	153	Лобанева Е.И.	90
Аверин А.В.	103	Ложкин А. Г.	119
Алехин Р.В.	136	Мазуров М.Ю.	103
Благосклонов Н.А.	47	Майоров К. Н.	119
Блашонков Т.О.	136	Марьясин О.Ю.	56
Бойкова М.А.	173	Махалова Т.П.	164
Боковой А.В.	127	Молодченков А.И.	145
Вагин В.Н.	182	Муравьев К.Ф.	127
Варшавский П.Р.	136	Мухачева И.Е.	136
Василькова М.А.	90	Наполи А.	164
Виноградов Д.В.	5	Окунь Д.Б.	81
Гельцер Б.И.	28	Панокин Н.В.	103
Грибова В.В.	47, 81	Петрова С.Ю.	173
Грицив Р.С.	28	Петряева М.В.	47, 81
Демикова Н.С.	47	Поляков С.А.	136
Дли М.И.	90	Пучков А.Ю.	90
Добрынин Д.А.	11	Рублев В.Ю.	28
Жевнерчук Д.В.	40	Сафроненкова И.Б.	69
Жуков Д.А.	19	Тулупьева Т.В.	153
Зубрихина М.О.	145	Фомина М.В.	182
Клячкин В.Н.	19	Шалфеева Е.А.	47, 81
Кобринский Б.А.	47	Шахгельдян К.И.	28
Корепанова А.А.	153	Яковлев К.С.	127
Костин И.А.	103		
Крашенинников В.Р.	19		
Крупецков А.В.	182		
Кувайскова Ю.Е.	19		
Кузнецов С.О.	164		
Кулик Б.А.	192		
Курейчик В.М.	69		

**АННОТАЦИИ СТАТЕЙ,
ОПУБЛИКОВАННЫХ В СБОРНИКЕ ТРУДОВ
КИИ-2019 ИЗДАТЕЛЬСТВОМ SPRINGER**

Секция 1

**ИСПОЛЬЗОВАНИЕ МЕТОДА РАНДОМИЗИРОВАННЫХ
СВОДНЫХ ПОКАЗАТЕЛЕЙ ДЛЯ ОЦЕНКИ УЩЕРБА ОТ
УСПЕШНЫХ СОЦИОИНЖЕНЕРНЫХ АТАКУЮЩИХ
ВОЗДЕЙСТВИЯ ЗЛОУМЫШЛЕННИКА**

А.А. Азаров (*artur-azarov@yandex.ru*)
Санкт-Петербургский институт информатики
и автоматизации РАН, Санкт-Петербург

О.В. Васильева (*vasiljevaa@mail.ru*)
Санкт-Петербургский государственный университет,
Санкт-Петербург

Т.В. Тулупьева (*tvt100a@mail.ru*)
Санкт-Петербургский институт информатики
и автоматизации РАН, Санкт-Петербург

В работе описываются применение метода рандомизированных сводных показателей при формировании оценки потенциального ущерба, который может быть нанесен компании при утечке конфиденциальной информации в случае социоинженерных атакующих воздействий злоумышленника. Данная оценка используется при сравнительном анализе эффективности различных мер, направленных на повышения уровня защищенности пользователей информационной системы от социоинженерных атакующих воздействий злоумышленника.

Ключевые слова: информационная безопасность, анализа защищенности пользователей информационных систем, социоинженерные атаки, метод рандомизированных сводных показателей

ТЕХНОЛОГИЯ ФОРМИРОВАНИЯ ПЕРСОНАЛИЗИРОВАННЫХ ПРОФИЛАКТИЧЕСКИХ РЕКОМЕНДАЦИЙ НА ОСНОВЕ ОЦЕНКИ РИСКОВ ЗАБОЛЕВАНИЙ

А.И. Молодченков (aim@tesyan.ru)
Федеральный исследовательский центр «Информатика
и управление» РАН, Москва
Российский университет дружбы народов, Москва

О.Г. Григорьев (olegpolikvart@yandex.ru)
Федеральный исследовательский центр «Информатика
и управление» РАН, Москва

В работе описывается технология формирования списка персонализированных профилактических рекомендаций. Технология содержит следующие основные компоненты: модуль получения данных о состоянии здоровья человека, базу данных, базу знаний, решатель с объяснением вывода. В представленной версии эта технология позволяет оценивать риски инсульта, инфаркта и депрессии и содержит более двухсот факторов рисков этих заболеваний и более двадцати профилактических рекомендаций. Обучение для этой версии проводилось на основе автоматизированного анализа большого числа публикаций и экспертных знаний.

Ключевые слова: здоровьесбережение, факторы риска, искусственный интеллект, базы знаний, неоднородные семантические сети, 4П медицина, профилактика, оценка рисков заболеваний

ОСОБЕННОСТИ ПОСТРОЕНИЯ КОМПЕТЕННОСТНО- ОРИЕНТИРОВАННЫХ МОДЕЛЕЙ СПЕЦИАЛИСТОВ НА ОСНОВЕ ТЕХНОЛОГИИ ОБУЧАЮЩИХ ИНТЕГРИРОВАННЫХ ЭКСПЕРТНЫХ СИСТЕМ

Г.В. Рыбина (galina@ailab.mephi.ru)
А.Ю.Никифоров (AYNikiforov@mephi.ru)
Е.С. Фонталина (deav@inbox.ru)
И.А.Сорокин (ziuvari@ya.ru)
Национальный исследовательский ядерный
университет «МИФИ», Москва

Анализируется полученный на кафедре «Кибернетика» НИЯУ МИФИ научно-методический и технологический опыт разработки и использования обучающих интегрированных экспертных систем и создание единого онтологического пространства знаний и умений для автоматизированного построения компетентностно-ориентированных моделей специалистов в области методов и технологий искусственного интеллекта по направлению подготовки «Программная инженерия».

Ключевые слова: искусственный интеллект, обучающие интегрированные экспертные системы, задачно-ориентированная методология, интеллектуальная программная среда, инструментальный комплекс АТ-ТЕХНОЛОГИЯ, интеллектуальное обучение, программная инженерия, модель обучаемого, технология программирования, компетентностно-ориентированная модель специалиста, онтологии курсов/дисциплин, профессиональные компетенции

Секция 2

ПЕРЕРАСПРЕДЕЛЕНИЕ АНИМАТОВ МЕЖДУ ГРУППАМИ

И.П. Карпова (*karpova_ip@mail.ru*)
Национальный исследовательский университет
«Высшая школа экономики»

В докладе рассматривается задача перераспределения роботов между группами. Описаны модели и механизмы, которые лежат в основе предложенного метода перераспределения. Приведено описание экспериментов по имитационному моделированию.

Ключевые слова: групповая робототехника, модели социального поведения, муравьи-рабовладельцы, агрессивное поведение, задача фуражировки

ИЕРАРХИЧЕСКАЯ АРХИТЕКТУРА УПРАВЛЕНИЯ РОБОТА ДЛЯ ОБУЧЕНИЯ С ГЕТЕРОГЕННЫМИ ПОВЕДЕНИЯМИ

М.А. Ровбо (*rovboma@gmail.com*)
А.Д. Московский (*moscowskyad@gmail.com*)
П.С. Сорокоумов (*petr.sorokoumov@gmail.com*)
НИЦ Курчатовский институт, Москва

В работе описывается архитектура управления для робототехнических систем с обучением, позволяющая комбинировать различные алгоритмы

целенаправленного поведения. Предлагается алгоритм управления верхнего уровня, переключающий управление между базовыми алгоритмами, основанными на разных подходах: Q-обучении, случайном блуждании, управлении на основе правил. Алгоритм реализован в виде программного модуля, работа которого проверена на примере задачи поиска заданной двери в здании сложной планировки. Проблема рассматривается как задача обучения с подкреплением в двух вариациях: с фиксированной целью между эпизодами и с меняющейся. Моделирование показало, что предложенный метод ведет себя более стабильно для различных вариантов задачи, чем каждый из базовых по отдельности, хотя и не дает наилучшего результата для каждого отдельного случая.

Ключевые слова: робот, архитектура управления, поведение, обучение с подкреплением

РЕАЛИЗАЦИЯ ТРАЕКТОРНОГО ДВИЖЕНИЯ БПЛА С ПРИМЕНЕНИЕМ ЛОКАЛЬНЫХ ИНТЕРПОЛИРУЮЩИХ СПЛАЙНОВ И ОПТИМИЗАЦИЕЙ УЗЛОВЫХ ТОЧЕК

М.В. Хачумов (*khmike@inbox.ru*)

Федеральный исследовательский центр «Информатика
и управление» РАН, Москва

В.М. Хачумов (*vmh48@mail.ru*)

Российский университет дружбы народов, Москва

Рассматривается задача отработки сложного траекторного движения беспилотного летательного аппарата (БПЛА) с оптимизацией расположения узлов интерполирующего сплайна. В качестве примеров рассмотрены типовые кривые второго порядка: эллипсы, параболы, гиперболы, получаемые в результате сечения конуса плоскостями. Предложены правила расстановки заданного числа узлов для сечений, заданных в параметрическом виде. Для воспроизведения кривых предлагается использование аппарата сплайн-интерполяции, который позволяет на единой методологической основе разрабатывать математическое обеспечение средств отработки сложных движений, хранить геометрическую информацию в компактном виде и воспроизводить траектории с наперед заданной точностью. При этом в качестве локальных сплайнов используются параметрический кубический сплайн Эрмита и кривые Безье, отличающиеся простотой вычислительных процедур. Проведены экспериментальные исследования поиска оптимального расположения узловых точек. В экспериментальной части статьи исследуется траекторное движение БПЛА по маршруту, представленному параболой, в условиях ветровой нагрузки с использованием математической модели летательного аппарата. При этом в

основе динамического планирования движения лежат стратегии и правила, имитирующие действия пилота в условиях необходимости принятия оперативных действий.

Ключевые слова: БПЛА, траектория движения, моделирование, конические сечения, локальный сплайн Эрмита, кривая Безье, узлы интерполяции, оптимизация

МОДЕЛИРОВАНИЕ СТРУКТУРЫ МИМО-АГЕНТОВ И МЕХАНИЗМОВ ИХ ВЗАИМОДЕЙСТВИЯ

Л.Ю. Жиликова (*zhilyakova.ludmila@gmail.com*)
Институт проблем управления РАН, Москва

В работе описана формальная модель пользователей социальной сети, имеющих некоторые наборы интересов разной тематики. Пользователи представлены гетерогенными агентами, имеющими множество входов разных типов и множество выходов разных типов (МИМО-агентами). Каждый тип соответствует одному из интересов пользователей. Агенты имеют кумулятивную функцию активации, зависящую как от внешних воздействий, так и от предыдущих состояний сети. Если значение этой функции на некотором такте выше заданного порога, агент становится активным по одному из типов. Тип, по которому активируется агент, зависит как от его внутренней структуры (личных предпочтений, заданных вектором), так и от доли активных соседей того или иного типа. Рассмотрены примеры распространения активности и показана зависимость устойчивых паттернов активности от параметров агентов. Сети МИМО-агентов со сходными свойствами могут применяться не только для описания взаимодействия пользователей социальных сетей, но и при моделировании процессов передачи разнородной информации в телекоммуникационных сетях.

Ключевые слова: сети сложных агентов, пороговые модели, МИМО-агенты, активность в сетях

ХАОТИЧЕСКИЕ ЯВЛЕНИЯ В КОЛЛЕКТИВНОМ ПОВЕДЕНИИ CHAOTIC PHENOMENA IN COLLECTIVE BEHAVIOR

В.Л. Стефанюк (*stefanuk@iitp.ru*)
Институт проблем передачи информации РАН, Москва

Т.О.Земцова (*zemcovato@yandex.ru*)
Российский университет дружбы народов, Москва

В статье изучается поведение коллектива из двух независимо управляемых когнитивных систем, описываемых связанной системой дифференциальных уравнений. Ранее для такого коллектива и его обобщений было показано, что вектор целей, стоящих перед системами, принадлежит некоторой области, которая разбивается на два региона. В первом регионе система допускает достижение поставленных целей в силу указанных уравнений. В настоящей работе показано, что во втором регионе, где выбранные цели не могут быть достигнуты, тоже можно говорить о коллективном поведении. Но последнее следует отнести к классу хаотических явлений. Это явление прежде не отмечалось, хотя оно может оказаться важным для некоторых практических приложений, поскольку обсуждаемые цели устанавливаются для системы извне или выбираются подсистемами коллектива независимо. Есть основания полагать, что подобным же образом описывается коллективное поведение пары обучающихся конечных автоматов с целесообразным поведением, при ограничении областью фазового пространства, в которой успешное совместное поведение таких автоматов невозможно.

Ключевые слова: хаотические явления в динамических системах, коллективное поведение, коллективное поведение интеллектуальных систем, технические системы, хаотическое поведение, граница хаоса.

Секция 4

АВТОМАТИЗАЦИЯ ВЫЯВЛЕНИЯ НАУЧНО-ТЕХНОЛОГИЧЕСКИХ ТРАЕКТОРИЙ

С.С. Волков

Российский Университет Дружбы Народов, Москва

Д.А. Девяткин

Федеральный исследовательский центр "Информатика и управление" Российской академии наук, Москва

И.В. Соченков

Федеральный исследовательский центр "Информатика и управление" Российской академии наук, Москва

И.А. Тихомиров

Министерство науки и высшего образования
Российской Федерации, Москва

Н. В. Тоганова
Институт Мировой Экономики и Международных
Отношений Российской Академии Наук, Москва

В статье предложен метод, позволяющий решить одну из подзадач выявления научно-технологических траекторий, а именно, автоматическое пополнение коллекции тематически схожих документов для дальнейшего анализа. Для решения данной задачи используются технологии анализа больших корпусов текстов. В работе представлено сравнение двух методов пополнения коллекции, а также анализ результирующей коллекции. Основная цель исследования – получить информацию о заданной технологии и о второстепенных технологиях, которые могли повлиять на ее развитие. В качестве информационного ресурса использовалась база патентных документов USPTO (более 4,4 млн. документов). В качестве примера была выбрана технология беспилотного автомобиля. В результате работы авторы пришли к выводу, что представленные методы могут быть использованы для решения поставленной задачи.

Ключевые слова: анализ текстов, научно-технологические траектории, беспилотные автомобили

МЕТОД ВЫЯВЛЕНИЯ РЕАЛИЗАЦИЙ КОГНИТИВНЫХ ОПЕРАЦИЙ В НАУЧНЫХ ТЕКСТАХ

Д.А. Девяткин
Федеральный исследовательский центр "Информатика
и управление" Российской академии наук, Москва

Теория риторических структур определяет отношения между клаузами и более крупными дискурсивными единицами, но абстрагируется от экстралингвистической природы создания текста. Однако процесс написания текста полностью связан с конкретной деятельностью, в рамках которой он проводился. В статье представлен новый подход, в котором текст рассматривается не как абстрактный продукт познавательной деятельности ученого, а моделируется сама познавательная деятельность, нашедшая отражение в научном тексте. В работе представлен метод выявления упоминаний когнитивных операций в научных текстах на русском языке, а также проводится его экспериментальная оценка. Полученные результаты показывают, что использованный набор когнитивных операций может быть использован для моделирования когнитивных процессов в научной деятельности, а также свидетельствуют о применимости предлагаемого метода для извлечения когнитивных операций из научных текстов на русском языке.

Ключевые слова: когнитивные операции, зоны аргументации, Random Forest

ОНТОЛОГО-УПРАВЛЯЕМАЯ ОБРАБОТКА НЕСТРУКТУРИРОВАННОГО ТЕКСТА

О.А. Невзорова (*onevzoro@gmail.com*)
Академия Наук Республики Татарстан, Казанский
федеральный университет, Казань

В.Н. Невзоров (*nevzorovvn@gmail.com*)
Казанский национальный исследовательский
технический университет им. А.Н. Туполева, Казань

Многие проекты по онтологиям направлены на описание некоторого аспекта реальности: объектов, отношений, состояний, событий и процессов в мире. Другой подход заключается в использовании онтологий для решения проблем. В этой статье мы обсуждаем подход к проектированию задач обработки текстов на основе многоуровневой системы онтологических моделей. Мы разработали систему онтологических моделей, которая используется для управления обработкой неструктурированных текстов. Компонентами онтологической системы являются онтология проектирования задач, онтология прикладных моделей и онтология предметной области. Обсуждается общая схема построения решений ряда прикладных задач, связанных с обработкой неструктурированных текстов.

Ключевые слова: онтология проектирования задач, онтология прикладных моделей, онтология предметной области, компьютерная обработка текстов

МЕТОДЫ ОПРЕДЕЛЕНИЯ ЛИЧНОСТНЫХ ЧЕРТ У ПОЛЬЗОВАТЕЛЕЙ СОЦИАЛЬНЫХ СЕТЕЙ

М. А. Станкевич (*stankevich@isa.ru*)
Институт проблем искусственного интеллекта ФИЦ
ИУ РАН, Москва

А. В. Латышев (*andrey.latyshhev@gmail.com*)
ООО «РИ Технологии», Москва

Н.В. Кисельникова (*nv.pirao@gmail.com*)
Психологический институт РАО, Москва

И. В. Смирнов (*ivs@isa.ru*)
Институт проблем искусственного интеллекта ФИЦ
ИУ РАН, Москва

Своевременное определение людей с риском развития психических и личностных расстройств крайне актуальна для современного общества. В большом количестве исследований показано, что личностные черты являются предикторами психических расстройств. В данной работе рассматривается задача автоматического определения личностных черт у пользователей социальной сети ВКонтакте. Описан набор данных профилей ВКонтакте, предложен набор признаков и методы решения задачи, представлены результаты экспериментов.

Ключевые слова: личностные черты, социальные сети, машинное обучение

ОДИН СПОСОБ АДАПТАЦИИ ПРЕДОБУЧЕННОЙ НЕЙРОСЕТЕВОЙ ЛИНГВИСТИЧЕСКОЙ МОДЕЛИ К ДОМЕНУ РАЗРАБОТКИ ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Дударин П. В. (*p.dudarin@ulstu.ru*)

Тронин В. Г. (*v.tronin@ulstu.ru*)

Святов К. В. (*k.svyatov@ulstu.ru*)

Ульяновский государственный технический университет,
Ульяновск

По данным CHAOS report from Standish Group на протяжении 1992-2017 годов степень успешности проектов в области разработки систем, интенсивно использующих программное обеспечение (Software Intensive Systems, SIS) изменилась значительно, оставаясь на уровне 50% несоответствия начальным требованиям (финансы, время и функциональность) для проектов средних размеров. Ежегодные финансовые потери в мире из-за полных провалов имеют порядок сотен миллиардов долларов. Большинство информации о проектах разработки ПО имеет текстовое представление. Анализ этой информации необходим для понимания общего состояния проекта и выявления проблем на ранних стадиях. Задачи обработки естественного языка в последнее время решаются с помощью нейросетевых лингвистических моделей. Эти модели показали уже превосходные результаты в классификации, переводе, распознавании именованных сущностей и ряде других задач. Предобученные модели доступны в сети Интернет, но в задачах, связанных со специфическими предметными областями, такими, как разработка программного обеспечения, не могут быть использованы в силу разности используемых словарей. Данная работа посвящена демонстрации способа адаптации предобученной нейросетевой лингвистической модели к домену разработки прикладного программного обеспечения.

Ключевые слова: NLP, обработка естественного языка, лингвистические модели, нейронные сети, дообучение сетей, нечеткие графы, кластеризация, RNN, ULMFiT, word-to-vec.

Секция 5

ЭВОЛЮЦИОННЫЙ ДИЗАЙН НЕЧЕТКИХ СИСТЕМ НА ОСНОВЕ МНОГОЦЕЛЕВОЙ ОПТИМИЗАЦИИ И ВЕРОЯТНОСТНЫХ СХЕМ ДЕМПСТЕРА-ШЕФФЕРА

А.И. Долгий

Ростовский филиал АО «НИИАС», Ростов-на-Дону

С.М. Ковалев (*ksm@rfniias.ru*)

Ростовский филиал АО «НИИАС», Ростов-на-Дону

А.Е. Колоденкова (*anna82_42@mail.ru*)

Самарский государственный технический университет,

Самара

А.В. Суханов

Ростовский филиал АО «НИИАС», Ростов-на-Дону

В статье разрабатывается новый интеллектуальный подход к проектированию нечетких систем на основе развития теории многоцелевого эволюционного моделирования Multiobjective Evolutionary Fuzzy Systems (MEFS). Разрабатываемый подход базируется на принципе Парето-оптимальности с привлечением вероятностных схем комбинирования свидетельств Демпстера-Шеффера. Схема комбинирования свидетельств используется в эволюционных операторах поисковых алгоритмов при реализации процедур назначения фитнесов и селекции решений. Для разрабатываемого класса многоцелевых нечетких систем разработаны новые формы представления интегрального и векторного критериев, отражающие как точностные и сложностные свойства разрабатываемых нечетких моделей, так и интерпретационные свойства, характеризующие удобочитаемость базы правил и семантическую непротиворечивость. Основным достоинством разрабатываемого класса многоцелевых нечетких систем и их главным конкурентным преимуществом является удовлетворение одновременно нескольким критериям, отражающим не только точность моделирования, но и интерпретационные свойства проектируемой нечеткой системы, такие как компактность описания, удобочитаемость, семантическая непротиворечивость, полнота описания. Предложен принципиально новый метод селекции и комбинирования решений на основе слияния оценок фитнесов нескольких индивидуумов с использованием вероятностной схемы комбинирования свидетельств Демпстера-Шеффера,

позволяющий выбирать из Парето-оптимальных наборов те решения, которые в наибольшей мере удовлетворяют многоцелевым установкам проектирования. Селекция и комбинирование решений на основе вероятностной теории объединения свидетельств повышают объективность выбора лучших решений в эволюционных алгоритмах. Разработаны новые методы ранжирования фитнесов решений в эволюционных алгоритмах и интеграции предпочтений экспертов в многоцелевой дизайн нечетких систем на основе модифицированных сетевых моделей комбинирования свидетельств Демпстера-Шеффера. Приводятся результаты сравнительных экспериментов по многоцелевому проектированию нечетких систем с использованием нескольких эволюционных алгоритмов на примере решения одной из задач в области железнодорожной автоматизации. Результаты экспериментов показали, что предложенный подход к проектированию нечетких систем обеспечивают лучший компромисс между интерпретируемостью и точностью, чем известные алгоритмы многоцелевого дизайна.

Ключевые слова: Многоцелевые нечеткие системы, схемы комбинирования свидетельств Демпстера-Шеффера, многокритериальность

ИЕРАРХИЯ ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИОННЫХ ОБЪЕКТОВ В СРЕДЕ ЭЛЕКТРОННОЙ БИБЛИОТЕКИ

Н.Е. Каленов (*nekalenov@mail.ru*)

И.Н. Соболевская (*nik_first@mail.ru*)

А.Н. Сотников (*asotnikov@jscs.ru*)

Межведомственный суперкомпьютерный центр Российской академии наук - филиал Федерального государственного учреждения «Федеральный научный центр Научно-исследовательский институт системных исследований Российской академии наук» (МСЦ РАН - филиал ФГУ ФНЦ НИИСИ РАН),
Москва

Формируемое цифровое пространство знаний является одним из важнейших элементов современного информационного общества. Предоставление доступа к объектам пространства знаний обеспечивается средствами сети интернет, где открываются широкие возможности объединения различных информационных источников, извлечения знаний и формирования на их основе виртуального информационного пространства. Эффективным средством интеграции информационных ресурсов является комплекс технологических, технических и организационных решений, объединенных понятием электронная библиотека, которая обеспечивает формирование и предоставление широкому кругу пользователей информационных ресурсов по различным направлениям. В работе

исследуется проблема построения современного информационного общества как задача формирования виртуального пространства знаний. Показано, каким образом средствами электронной библиотеки могут быть интегрированы электронные копии объектов библиотечного, архивного и музейного хранения, представленные в виде текстов, графических образов, аудио-видео объектов, включая 3D-модели. Введено понятие уровня иерархии электронных объектов. Даны определения объектов различных уровней и сформулированы принципы работы с объектами каждого из уровней. Предложена иерархия представления электронных объектов в среде электронной библиотеки.

Keywords: электронная библиотека «Научное наследие России», тематико-видовые коллекции, тематические коллекции, междисциплинарные коллекции, виртуальные выставки, цифровые 3D-модели, музейные объекты

СРАВНЕНИЕ ПОЛИ ИНТЕРВАЛЬНЫХ АЛЬТЕРНАТИВ: ОЦЕНКА КОЛЛЕКТИВНОГО РИСКА

Г.И. Шепелев (gis@isa.ru)

Федеральный исследовательский центр «Информатика
и управление» РАН, Москва

Для поли интервальных объектов, обобщенных интервальных и нечетких величин, в рамках метода оценки коллективного риска для случая попарного сравнения предложены процедуры расчета индикаторов предпочтительности и риска, ранее применявшиеся к моно интервальным объектам. Процедуры основаны на дефаззификации интервальных оценок индикаторов предпочтительности и риска, относящихся к моноинтервалам на альфа-уровнях в случае нечетких поли интервальных объектов, и на представлении обобщенных интервальных оценок в виде вероятностной смеси образующих такую оценку распределений. Исследуются некоторые различия подходов обобщенных интервальных оценок и нечетких оценок при сравнении альтернатив, а также связь указанных подходов. Установлено, что обобщенные равномерные распределения шансов в подходе обобщенных интервальных оценок получаются при дефаззификации равномерных распределений на альфа-уровнях нечетких объектов. Показано, как процедуры дефаззификации приводят к одно-числовым оценкам интервальных характеристик нечетких объектов, аналогичных числовым характеристикам функций распределения теории вероятностей (математическое ожидание, дисперсия, среднее полуотклонение). В зависимости от способа дефаззификации из равномерных распределений на альфа-уровнях нечетких объектов могут быть получены различные распределения шансов в формализме обобщенных интервальных оценок. Получаемыми при этом распределениями, однако, не исчерпывается все многообразие распределений шансов, возникающих в последнем формализме.

Ключевые слова: сравнение поли интервальных альтернатив, обобщенные интервальные оценки, нечеткие поли интервальные альтернативы, методы дефаззификации, одно-числовые оценки интервальных характеристик нечетких объектов, метод оценки коллективного риска, процедуры сравнения поли интервальных альтернатив

Секция 6

НЕЙРОСЕТЕВОЙ МЕХАНИЗМ КВАНТОВО-ПОДОБНЫХ КОГНИТИВНЫХ ФЕНОМЕНОВ НА АЛГЕБРЕ ФУРЬЕ- ДУАЛЬНЫХ ОПЕРАЦИЙ ПРИ ЛИНЕЙНОЙ РЕГИСТРАЦИИ ВЕСОВ СВЯЗЕЙ

А.В. Павлов (*pavlov@phoi.ifmo.ru*)
Университет ИТМО, Санкт-Петербург

Показан механизм квантово-подобного когнитивного феномена «Линда» на нейросети, порождающей алгебру фурье-дуальных операций. Ключевые факторы механизма суть субъективность внутренних репрезентаций и утомляемость синаптической передачи. Модель свободна от требования на некоммутативность операторов и актуальна даже при линейной записи весов межнейронных связей.

Ключевые слова: квантово-подобные феномены, когнитивный феномен, квантовая логика, нейронные сети, ассоциативная память, нелинейность, фурье-дуальность, активационная функция нейрона

Секция 7

СБОР ДАННЫХ И ПОДГОТОВКА ОБУЧАЮЩЕЙ ВЫБОРКИ ДЛЯ ЗАДАЧИ ДИАГНОСТИКИ ПАТОЛОГИЙ ЗРЕНИЯ

А.П. Еремеев (*eremeev@appmat.ru*)
Национальный исследовательский университет
«МЭИ», Москва
С.А. Ивлиев (*siriusfrk@gmail.com*)
Национальный исследовательский университет
«МЭИ», Москва

В работе представлены подход и методика, которые были использованы для сбора и хранения обучающей выборки, а также её предварительной обработки с целью последующего использования в системе поддержки принятия решений для диагностики сложных патологий зрения. Обучающая выборка состоит из различных записей биопотенциалов, полученных со специального медицинского аппарата.

Ключевые слова: поддержка принятия решений, диагностика, обучающая выборка, размеченные данные, цифровая обработка сигналов, пользовательский интерфейс.

ИЕРАРХИЧЕСКОЕ ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ С ИСПОЛЬЗОВАНИЕМ КЛАСТЕРИЗУЮЩИХ АБСТРАКТНЫХ АВТОМАТОВ

А.Скрынник(*skrynnik@isa.ru*)

Институт проблем искусственного интеллекта ФИЦ
“Информатика и Управление” Российской Академии
Наук, Москва

А.И. Панов (*pan@isa.ru*)

Институт проблем искусственного интеллекта ФИЦ
“Информатика и Управление” Российской Академии
Наук, Москва

Московский физико-технический институт, Москва

Иерархическое обучение с подкреплением (HRL) является еще одним шагом к сближению методов обучения и планирования. Составление абстрактных планов упрощает перенос знаний, а также способствует стабильности обучения агента в сложных окружениях с отложенным вознаграждением. Однако, для практического применения методов HRL, особенно в робототехнике, существует ряд трудностей, среди которых ключевой является задача создания иерархии действий без использования вручную добавленной информации, которую агент использует в качестве предварительно обучаемой схемы. В данной статье мы предлагаем подход для одновременного построения и применения иерархии действий и подцелей. В отличие от предыдущих работ в этом направлении, метод основан на едином цикле кластеризации состояний окружения и выявления подцелей с помощью модифицированного подхода для поиска «узких мест», для построения иерархии абстрактных автоматов. Общий автомат строится с использованием программируемых схем, которые являются достаточно универсальными для организации переноса знаний и применимы для широкого класса задач. Каждому кластеру состояний окружения назначается отдельный автомат, который является частью общего автомата. Целью каждого автомата является достижение состояния

«узкого места», а затем переход в другой кластер. Мы оцениваем наш подход, используя усложненный набор экспериментов в задаче планирования.

Ключевые слова: машинное обучение, обучение с подкреплением, иерархическое обучение с подкреплением

АЛГЕБРАИЧЕСКИЕ БАЙЕСОВСКИЕ СЕТИ: ЛОКАЛЬНОЕ МАШИННОЕ ОБУЧЕНИЕ НА ОСНОВЕ НЕСОВЕРШЕННОЙ ИНФОРМАЦИИ ИЗ СОЦИАЛЬНЫХ СЕТЕЙ

А.Г. Максимов (*maksimov.20.43@gmail.com*)

Санкт-Петербургский институт информатики и
автоматизации Российской академии наук, Санкт-
Петербург

А.Л. Тулупьев (*alt@dscs.pro*)

Санкт-Петербургский государственный университет,
Санкт-Петербург

Н.А. Харитонов (*nikita.kharitonov95@yandex.ru*)

Санкт-Петербургский институт информатики и
автоматизации Российской академии наук, Санкт-
Петербург

Одной из задач, решаемых в рамках работы с моделями машинного обучения, является задача собственно обучения модели на имеющихся данных. В частности, эта задача возникает при работе с алгебраическими байесовскими сетями. Ранее были описаны подходы к обучению последних, данное исследование посвящено алгоритму обучения алгебраических байесовских сетей, представленных в виде фрагмента знаний, на данных с пропусками. Кроме предложенного алгоритма продемонстрирован пример его работы как на искусственных данных, так и на реальных данных из социальных сетей.

Ключевые слова: машинное обучение, алгебраические байесовские сети, неполная информация, пропущенные значения, социальные сети, алгоритм обучения

Секция 8

ОНТОЛОГИЧЕСКИЙ ПОДХОД К ОБЕСПЕЧЕНИЮ ИНТЕЛЛЕКТУАЛЬНОЙ ПОДДЕРЖКИ РЕШЕНИЯ ВЫЧИСЛИТЕЛЬНО СЛОЖНЫХ ЗАДАЧ НА СУПЕРКОМПЬЮТЕРАХ

Г.Б. Загорулько (*gal@iis.nsk.su*),
Ю.А. Загорулько (*zagor@iis.nsk.su*)
Институт систем информатики СО РАН им. А.П.
Ершова, Новосибирск
Б.М. Глинский (*gbm@opg.sbcc.ru*),
А.Ф. Сапетина (*afsapetina@gmail.com*)
Институт вычислительной математики и
математической геофизики СО РАН

В работе описывается подход к обеспечению интеллектуальной поддержки решения вычислительно сложных задач на суперкомпьютерах, базирующийся на онтологии и правилах принятия решений, помогающих пользователю выбрать оптимальные для его задачи вычислительный метод и архитектуру суперЭВМ. Рассматриваются концепция и следующие компоненты интеллектуальной поддержки: онтология проблемной области «Решение вычислительно сложных задач математической физики на суперкомпьютерах», информационно-аналитический интернет-ресурс, базирующийся на данной онтологии и обеспечивающий пользователю доступ к информации, необходимой для решения вычислительно сложных задач на суперкомпьютерах, и экспертная система, помогающая пользователям разрабатывать параллельный код на основе готовых программных компонент. Подробно описываются концептуальная схема интеллектуальной поддержки и разрабатываемые онтологии. Для исключения возможных ошибок при создании онтологии, моделирования дополнительных средств представления знаний, отсутствующих в языке описания онтологий, систематизации и упрощения пополнения онтологии экземплярами понятий был разработан ряд структурно-содержательных паттернов.

Ключевые слова: суперкомпьютеры, интеллектуальная поддержка вычислений, онтология, математическая физика, системы поддержки принятия решений

ОРГАНИЗАЦИЯ ИНФОРМАЦИОННОГО ВЗАИМОДЕЙСТВИЯ МЕЖДУ РЕЛЯЦИОННЫМИ БАЗАМИ ДАННЫХ С ПРИМЕНЕНИЕМ МЕТОДОВ ОНТОЛОГИЧЕСКОГО ИНЖИНИРИНГА

Н.Г. Ярушкина (*jng@ulstu.ru*)
А.А. Романов (*romanov73@gmail.com*)
А.А. Филиппов (*al.filippov@ulstu.ru*)
Ульяновский государственный технический
университет, Ульяновск

В статье описан пример консолидации данных между двумя реляционными базами данных (РБД). Предложенный подход заключается в применении методов онтологического инжиниринга для извлечения онтологий из метасхем РБД и их слияния для последующей организации информационного взаимодействия между этими РБД. Показана разница между традиционным и предложенным алгоритмами консолидации данных, рассмотрены их преимущества и недостатки.

Ключевые слова: реляционные базы данных, метасхема, онтология, слияние онтологий, информационное взаимодействие

Секция 9

МНОЖЕСТВЕННАЯ КЛАССИФИКАЦИЯ НА ОСНОВЕ СВЕРТОЧНЫХ НЕЙРО-НЕЧЕТКИХ СЕТЕЙ

В.В. Борисов (*vbor67@mail.ru*)
К.П. Коршунова (*ksenya-kor@mail.ru*)
Филиал ФГБОУ ВО «Национальный исследовательский
университет «МЭИ» в г. Смоленске

В статье рассмотрено решение задачи множественной классификации изображений на основе предложенной разновидности сверточных нейро-нечетких сетей, которая, помимо сверточных и субдискретизирующих слоев, перед полносвязными слоями нейронов включает в себя слой нечетких нейронов для кластеризации. Такие сети сочетают возможности сверточных нейронных сетей и особенности нечеткой кластеризации. Они позволяют более точно решать задачи множественной классификации изображений сложных объектов и сцен, которые характеризуются неопределенностью границ между неизолрованными и пересекающимися классами.

Ключевые слова: множественная классификация, нечеткая кластеризация, сверточные нейро-нечеткие сети.

ПРОГНОЗИРОВАНИЕ ЦЕН НА АКЦИИ С ПОМОЩЬЮ LSTM СЕТЕЙ

Н.К. Андриевская (*nataandr@yandex.ua*)

Т.А. Васяева (*vasyaeva@gmail.com*)

Т.В. Мартыненко (*tatyana.v.martynenko@gmail.com*)

С.В. Хмелевой (*hmelevoy_serгей@ukr.net*)

Донецкий национальный технический университет, Донецк

Рассмотрено применение глубоких LSTM нейронных сетей для прогнозирования цен акций сети аптек «36 и 6». Показано формирование обучающей выборки в задачах машинного обучения применительно к временным рядам. Предложена архитектура нейронной сети. Разработана и обучена нейронная сеть для прогнозирования биржевой торговли с использованием языка программирования Python, а также библиотеки Keras. Проведены эксперименты по настройке основных параметров.

Ключевые слова: машинное обучение, глубокое обучение, рекуррентные нейронные сети, LSTM, временной ряд, цены на акции.

ПРИМЕНЕНИЕ НЕЙРОННОЙ СЕТИ ХЕММИНГА ДЛЯ ПРОГНОЗИРОВАНИЯ ПРОЧНОСТИ СВАРНЫХ СОЕДИНЕНИЙ

В.С. Климов (*klimovv@gmail.com*)

А.С. Климов (*klimovTGU@yandex.ru*)

С.В. Мкртычев (*sm5006@yandex.ru*)

Тольяттинский государственный университет, Тольятти

В статье рассматривается применение нейронной сети Хемминга для прогнозирования предельной силы разрушения при нагрузке сварного соединения на срез. Представлены результаты обучения и тестирования нейронной сети. Как показал анализ, относительная погрешность прогнозирования силы разрушения сварного шва не превышает 10 %.

Ключевые слова: нейронная сеть Хемминга, прогнозирование, контактная сварка, прочность сварного соединения

Секция 10

ЛОГИЧЕСКАЯ КЛАССИФИКАЦИЯ ЧАСТИЧНО УПОРЯДОЧЕННЫХ ДАННЫХ

Е.В. Дюкова (*edjukova@mail.ru*)

ФИЦ ИУ РАН, Москва

Г.О. Масляков (*gleb-mas@mail.ru*)

МГУ им. М.В. Ломоносова, Москва

П.А. Прокофьев (*p_prok@mail.ru*)

ИМАШ РАН, Москва

В статье исследованы вопросы интеллектуального анализа данных, возникающие при решении задач машинного обучения. Предложена схема синтеза корректных процедур классификации по прецедентам, ориентированных на задание отношений частичных порядков на множествах значений признаков и базирующихся на обобщении классических понятий логической классификации. Показано, что в общем случае на этапе обучения корректного логического классификатора возникает необходимость рассматривать одну из центральных труднорешаемых дискретных задач, а именно, задачу дуализации над произведением частичных порядков. Дана матричная формулировка дуализации над произведением частичных порядков. Эффективность предлагаемого подхода к задаче классификации по прецедентам проиллюстрирована на модельных и реальных данных.

Ключевые слова: логический анализ данных, классификация по прецедентам, монотонная дуализация, дуализация над произведением частичных порядков, неприводимое покрытие булевой матрицы, упорядоченное тупиковое покрытие целочисленной матрицы

ПРОСТРАНСТВЕННАЯ КЛАСТЕРИЗАЦИЯ НА ОСНОВЕ АНАЛИЗА БОЛЬШИХ ДАННЫХ ДЛЯ ЦИФРОВОГО МАРКЕТИНГА

А.В. Ивашенко (*anton.ivashenko@gmail.com*)

Самарский государственный технический университет,

Самара

А.А. Столбова (*anastasiya.stolbova@bk.ru*),

О.К. Головнин (*golovnin@bk.ru*)

Самарский национальный исследовательский
университет имени академика С.П. Королева, Самара

Анализ и визуализация больших объемов слабоструктурированной информации (Big Data) в задачах поддержки принятия решений является важной и актуальной проблемой цифровой экономики. Данная статья посвящена решению этой проблемы в области цифрового маркетинга, например, для размещения торговых точек и центров предоставления услуг в условиях города. Предлагается адаптивный подход к формированию пространственных сегментов урбанизированной территории на основе анализа и областей спроса и предложения и их визуализации на электронной карте местности. Предлагаемый подход к согласованию спроса и предложения включает в себя 3 этапа: семантико-статистический анализ, позволяющий выстроить зависимости между объектами, порождающими спрос, автоматизированный поиск баланса между спросом и предложением и визуализацию вариантов решения. Предлагаемый подход может быть использован для решения задач оптимизации сетей медицинских учреждений, торговых и культурно-развлекательных центров, социальных служб.

Ключевые слова: сегментация, кластеризация, большие данные, геомаркетинг, цифровая экономика.

**ТЕМАТИЧЕСКОЕ КАРТОГРАФИРОВАНИЕ И ОЦЕНКА
ВРЕМЕННЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ
МНОГОЗОНАЛЬНЫХ СПУТНИКОВЫХ ИЗОБРАЖЕНИЙ**

В.Е. Дементьев (*dve@ulntc.ru*)

А.Г. Френкель (*j.skvoll@gmail.com*)

Д.С. Кондратьев (*kondratev.dmitriy@gmail.com*)

А.С. Стрельцова (*nastya94strel@mail.ru*)

Ульяновский государственный технический университет,
Ульяновск

В работе предлагаются и сравниваются алгоритмы тематического картографирования многозональных спутниковых снимков. Предлагается нелинейный многомерный фильтр, позволяющий объединять результаты обработки нескольких разновременных многозональных изображений. Рассматриваются варианты реализации представленных процедур.

Ключевые слова: дистанционное зондирование Земли, спутниковый мониторинг, обработка последовательности изображений, метод наименьших квадратов, сегментация.

Секция 11

РАСШИРЕННЫЕ ШАГОВЫЕ ТЕОРИИ АКТИВНОЙ ЛОГИКИ: ДЕКЛАРАТИВНАЯ СЕМАНТИКА

М. Виньков (*vinkovmm@mail.ru*)

Государственный технический университет им.

Н.Э.Баумана, Москва

И.Б. Фоминых (*igborfomin@mail.ru*)

Национальный исследовательский университет

«МЭИ», Москва

Активная логика является концептуальной системой, принципам которой удовлетворяют формализмы рассуждений, позволяющие соотносить результаты рассуждений с моментами времени и обеспечивающие толерантность к противоречиям. В настоящее время толерантность к противоречиям (паранепротиворечивость) применительно к системам Активной Логики получила теоретическое обоснование в работах авторов данного доклада и относится к т.н. формализмам шаговых теорий, объединяющим в себе принципы Активной Логики и логического программирования. В частности, было доказано, что паранепротиворечивой является аргументационная семантика формализма т.н. расширенных шаговых теорий, имеющих два вида отрицания. Этот формализм обладает более развитыми выразительными возможностями по сравнению с другими формализмами шаговых теорий Активной Логики и более полно по сравнению с ними отвечает принципам логического программирования. В данной работе для формализма расширенных шаговых теорий Активной Логики предлагается декларативная семантика и показывается её эквивалентность существующей аргументационной семантики данного формализма. Это, в свою очередь, означает, что предложенная декларативная семантика также является паранепротиворечивой, а наличие логических противоречий в расширенных шаговых теориях не приводит к их разрушению.

Ключевые слова: активная логика, рассуждения во времени, шаговые теории, декларативная семантика, паранепротиворечивость.

ОРГАНИЗАЦИЯ ПОДДЕРЖКИ АНАЛИЗА АРГУМЕНТАЦИИ В НАУЧНО-ПОПУЛЯРНОМ ДИСКУРСЕ НА ОСНОВЕ ОНТОЛОГИЧЕСКОГО ПОДХОДА

Ю.А. Загорулько (*zagor@iis.nsk.su*)

Н.О. Гаранина (*garanina@iis.nsk.su*)

А.С. Серый (*alexey.seryj@iis.nsk.su*)
Институт систем информатики им. А.П. Ершова СО
РАН, Новосибирск
О.А. Доманов (*odomanov@gmail.com*)
Институт философии и права СО РАН, Новосибирск

В работе представлен подход к моделированию и анализу аргументации, встречающейся в научно-популярной литературе. Моделирование аргументации выполняется средствами онтологии аргументации, основанной на базовой онтологии AIF, расширенной средствами, позволяющими моделировать целевую аудиторию и более детально описывать содержание аргументов. В терминах этой онтологии описываются схемы аргументации, структура и элементы аргументов, а также могут представляться сети аргументов и их составные части, извлеченные из исследуемых текстов. Для анализа аргументации разрабатывается программный комплекс, который предоставляет средства для моделирования и выявления структуры аргументации на корпусе текстов, относящихся к научно-популярному дискурсу. Кроме того, программный комплекс предоставляет средства для исследования обнаруженной аргументации с целью выявления и анализа используемых в научных и научно-популярных текстах аргументативных стратегий и риторических приемов. В работе описаны особенности предлагаемой онтологии аргументации и представлены архитектура и функциональные возможности разрабатываемого программного комплекса.

Ключевые слова: аргументация, схема аргументации, средства анализа аргументации, онтология, научно-популярный дискурс

МЕТОДЫ ВЫВОДА ДЛЯ ОДНОГО КЛАССА СИСТЕМ ПРИ МНОГИХ НЕЧЁТКИХ ВХОДАХ

В.Г. Синюк (*vgsinuk@mail.ru*)
М.В. Панченко (*panchenko.maks@gmail.com*)
Белгородский государственный технологический
университет им В.Г. Шухова, Белгород

В статье разработаны методы вывода для нечётких систем типа Мамдани, которые при многих нечётких входах и при любых t-нормах реализуются с полиномиальной вычислительной сложностью. Для базы правил на этапе дефаззификации применены методы среднего центра и дискретный вариант центра тяжести. Приведены сетевые структуры систем, соответствующих этим методам.

Ключевые слова: системы с нечеткими входами, нечеткое значение истинности, метод вывода Мамдани, мера возможности

РАЗРАБОТКА ЛЕКСИКОНА ИНДИКАТОРОВ АРГУМЕНТАЦИИ

И.С. Кононенко (*irina_k@cn.ru*)

Е.А. Сидорова (*lsidorova@iis.nsk.su*)

Институт систем информатики им. А.П. Ершова
СО РАН, Новосибирск

В работе представлены результаты предварительного анализа индикаторов аргументации, обнаруженных при разметке корпуса русскоязычных научно-популярных текстов. Выявлены основные прагматические аспекты аргументации, о которых могут сигнализировать дискурсивные показатели. Наряду с прагматическими значениями, классификация индикаторов учитывает тип используемых языковых средств. Особое внимание уделено малоизученным индикаторным конструкциям и классам знаменательных слов. Рассмотрены конструкции с глаголами и существительными с семантикой ментального состояния и ментального воздействия, речи и вывода. Индикаторы аргументации представляются в виде лексических единиц и лексико-грамматических шаблонов, которые автоматически формируются по размеченным фрагментам текста, а затем при необходимости уточняются экспертом вручную. Язык описания шаблонов включает средства представления грамматических и семантических ограничений, вложенных шаблонов и альтернатив, разрывных конструкций.

Ключевые слова: индикатор аргументации, лексико-грамматический шаблон, аргументативная разметка текста

Содержание

Секция 7 Машинное обучение

<i>Д.В. Виноградов</i> Вычисление сходств обучающих примеров для вкф-метода на видеокάρтах	5
<i>Д.А. Добрынин</i> Переобучение в ДСМ-системах	11
<i>Д.А. Жуков, В.Н. Клячкин, В.Р. Крашенинников, Ю.Е. Кувайскова</i> Обновление агрегированных классификаторов при поступлении новой информации о показателях функционирование объекта	19
<i>Б.И. Гельцер, Р.С. Грицив, В.Ю. Рублев, К.И. Шахгельдян</i> Сравнительная оценка методов машинного обучения в прогнозировании отдаленных результатов аортокоронарного шунтирования	28

Секция 8 Инженерия знаний и онтологии

<i>Д.В. Жевнерчук</i> Модификация алгоритма сопряжения компонентов открытых информационных систем	40
<i>Б.А. Кобринский, Н.А. Благосклонов, Н.С. Демикова, В.В. Грибова, Е.А. Шалфеева, М.В. Петряева</i> Возможности применения онтологического подхода к диагностике орфанных заболеваний	47
<i>О.Ю. Марьясин</i> Онтология инженерных систем зданий для цифрового строительства.....	56
<i>В.М. Курейчик, И.Б. Сафроненкова</i> Поддержка функционирования распределенных сапр в «туманных» средах	69
<i>В.В. Грибова, Д.Б. Окунь, М.В. Петряева, Е.А. Шалфеева</i> Инфраструктура IASRAAS для формирования интерпретируемых баз диагностических знаний по заболеваниям произвольной направленности.....	81

Секция 9 Нейросетевые технологии

<i>М.И. Дли, А.Ю. Пучков, Е.И. Лобанева, М.А. Василькова</i> Выбор архитектуры глубоких нейронных сетей для отслеживания динамики состояния объекта	90
---	----

<i>А.В. Аверин, И.А. Костин, М.Ю. Мазуров, Н.В. Панокин</i> Анализ радиолокационных данных с помощью сверточных нейронных сетей для задачи классификации агентов трехмерной дорожной сцены.....	103
<i>А. Г. Ложкин, К. Н. Майоров</i> Применение механизма симметрий в нейронных сетях.....	119

Секция 10 Интеллектуальный анализ данных

<i>А.В. Боковой, К.Ф. Муравьев, К.С. Яковлев</i> Восстановление карт глубин изображений в задаче одновременного картирования и локализации по видеопотоку на встраиваемой системе NVIDIA JETSON TX2	127
<i>П.Р. Варшавский, Р.В. Алехин, Т.О. Блашонков, И.Е. Мухачева, С.А. Поляков</i> Разработка модульной прецедентной системы для решения задач классификации данных	136
<i>А.И. Молодченков, М.О. Зубрихина</i> Картина мира агента-онколога для решения задачи диагностики рака поджелудочной железы.....	145
<i>А.А. Корепанова, М.В. Абрамов, Т.В. Тулупьева</i> Идентификация аккаунтов пользователей в социальных сетях «ВКонтакте» и «Одноклассники»	153
<i>Т.П. Махалова, С.О. Кузнецов, А. Наполи</i> Майнинг множеств признаков на основе сжатия: вероятностный подход	164
<i>С.Ю. Петрова, М.А. Бойкова</i> Пример реализации рекомендательной системы на основе неявных оценок.....	173

Секция 11 Моделирование рассуждений

<i>В.Н. Вагин, А.В. Крупецков, М.В. Фомина</i> Алгоритм построения деревьев решений при наличии противоречий в данных	182
<i>Б.А. Кулик</i> Исследование противоречий в естественных рассуждениях на примерах метафор и пресуппозиций	192
Аннотации статей, опубликованных в сборнике трудов КИИ-2019 издательством Springer.....	202

Научное издание

**Семнадцатая Национальная конференция
по искусственному интеллекту с международным участием.
КИИ-2019**

(21–25 октября 2019 г., г. Ульяновск, Россия).

Сборник научных трудов.

Ответственный за выпуск - Н.Г. Ярушкина

ЛР № 020640 от 22.10.97

Подписано в печать 16.10.2019. Формат 60×84/16.

Усл. п. л. 13,25. Тираж 175 экз. Заказ 929.

Ульяновский государственный технический университет
432027, г. Ульяновск, ул. Сев. Венец, д. 32

ИПК «Венец» УлГТУ, 432027, г. Ульяновск, Сев. Венец, 32.

Научное электронное издание

Дата подписания к использованию 25.10.2019.
ЭИ № 1355. Объем данных 8,6 Мб. Заказ № 929.

Ульяновский государственный технический университет
432027, Ульяновск, Сев. Венец, 32.
ИПК «Венец» УлГТУ, 432027, Ульяновск, Сев. Венец, 32.

Тел.: (8422) 778-113
E-mail: venec@ulstu.ru
venec.ulstu.ru