

Нейроинформатика, её приложения и анализ данных

XXVIII Всероссийский семинар

**Красноярск
2020**

Министерство науки и высшего образования РФ
Сибирское отделение РАН
Российская ассоциация нейроинформатики
Институт вычислительного моделирования СО РАН

НЕЙРОИНФОРМАТИКА, ЕЁ ПРИЛОЖЕНИЯ И АНАЛИЗ ДАННЫХ

МАТЕРИАЛЫ
ДВАДЦАТЬ ВОСЬМОГО ВСЕРОССИЙСКОГО СЕМИНАРА
25 сентября – 27 сентября 2020 года

Красноярск 2020

УДК 007.52 (042.3)

Н 43

Нейроинформатика, её приложения и анализ данных: Материалы XXVIII Всероссийского семинара, 25 сентября – 27 сентября 2020 года / Под ред. М.Г. Садовского, отв. за вып. М.Ю. Сенашова; – Красноярск: Институт вычислительного моделирования СО РАН, 2020. – 151 с.

В сборнике представлены материалы XXVIII Всероссийского семинара «Нейроинформатика, её приложения и анализ данных», проходившей в городе Красноярске 25 сентября – 27 сентября 2020 года.

Основной задачей конференции является всесторонний и высококвалифицированный обмен новейшими достижениями в различных областях нейроинформатики (как теории нейросетей, так и в области различных практических приложений), а также в области нелинейного статистического анализа многомерных данных, обладающих нетривиальными структурами.

Большое внимание уделено анализу областей применимости и точности методов обработки многомерных данных, анализу устойчивости различных новых (нелинейных) методов кластеризации, разбор большого числа конкретных случаев, иллюстрирующих эти проблемы и достижения.

Материалы предназначены для научных работников, преподавателей, студентов и аспирантов соответствующих специальностей.

Конференция проводится при поддержке Красноярского математического центра, финансируемого Минобрнауки РФ в рамках мероприятий по созданию и развитию региональных НОМЦ (Соглашение 075-02-2020-1631).

Редакционная коллегия:

Садовский Михаил Георгиевич – ответственный редактор

Сенашова Мария Юрьевна – ответственный за выпуск

© ИВМ СО РАН, 2020

© Коллектив авторов, 2020

ISBN 978-5-6043558-7-9



9 785604 355879

АРХИТЕКТУРА ПРОГРАММНОЙ ПЛАТФОРМЫ ДЛЯ РАЗРАБОТКИ СРЕДСТВ МОДЕЛИРОВАНИЯ РАСТУЩИХ СПАЙКОВЫХ НЕЙРОННЫХ СЕТЕЙ¹

А.В.Бахшиев^{1,2}, И.С. Фомин¹, Ф.В.Гунделах^{1,2}, А.А. Демчева², А.М. Корсаков¹

¹ Центральный научно-исследовательский и опытно-конструкторский институт робототехники и технической кибернетики (ЦНИИ РТК), alexab@rtc.ru,

² Санкт-Петербургский политехнический университет Петра Великого (СПбПУ)

Введение

Одним из актуальных направлений исследований в области искусственного интеллекта является разработка новых, биологически-инспирированных, архитектур нейронных сетей, базирующихся на спайковых моделях нейронов. Основным методом исследования свойств таких сетей является численное моделирование. На сегодняшний день реализован ряд симуляторов спайковых нейронных сетей. Разработка кроссплатформенного симулятора NEST была начата в 1993 году [1]. Симулятор оптимизирован для расчёта больших нейрональных сетей, состоящих из сравнительно простых однокомпарментных нейронов. В настоящее время NEST способен симулировать модель, состоящую из $1,73 \cdot 10^9$ нейронов и приблизительно 10^{13} связей между ними. Большинство реализованных моделей нейрона состоит из небольшого количества компарментов, что снижает степень детализации описываемых биофизических процессов и морфологических особенностей отдельной клетки. Нейроны образуют сеть посредством так называемых синаптических контактов, для каждого из которых может быть определена собственная динамика. Простейшая модель связи между узлами характеризуется весом (определяет силу взаимодействия между узлами) и временем задержки (время, требуемое на переход сигнала от одного узла к другому). В симуляторе встроены модели синапсов с реализованными механизмами пластичности, синаптической депрессии и восстановления, а также предусмотрены функции для создания различных топологических схем связей и структурирования больших сетей.

Симулятор GENESIS был разработан для работы с биологически реалистичными симуляциями различных уровней — от субклеточных компонентов и биохимических реакций до сложных моделей нейрона и больших сетей, а также моделей системного уровня [2]. Симуляции в GENESIS строятся из отдельных модулей, которые принимают входные сигналы, производят внутри некоторые вычисления и генерируют выходной сигнал. Модели нейронов собираются из отдельных базовых компонентов — компарментов и ионных каналов переменной ёмкости. Компарменты связаны с соответствующими каналами и друг с другом, формируя таким образом многокомпонентную модель нейрона любого желаемого уровня сложности. Нейроны, в свою очередь, могут быть соединены друг с другом в сети [3].

Имитационная среда NEURON предназначена для моделирования отдельных нейронов и сетей нейронов [3]. Наряду с GENESIS, Neuron является основой для обучения вычислительной нейробиологии во многих курсах и лабораториях по всему миру. Neuron

¹ Работа проводилась в рамках выполнения государственного задания Минобрнауки России № 075-01195-20-00 “Разработка и исследование новых архитектур реконфигурируемых растущих нейронных сетей, методов и алгоритмов их обучения”

имеет графический пользовательский интерфейс (GUI), предназначенный для использования с минимальной необходимостью программирования. При помощи базовых моделей из клеток, каналов и сетей. можно сформировать основу для всех симуляций и моделей.

BRAIN – одна из наиболее развитых программных сред, позволяющая, в том числе, описывать собственные модели в дифференциальных уравнениях, а также обеспечивающая развитые средства интеграции с другими средами моделирования [4]. В отличие от NEST и GENESIS, позволяющих создавать мультикомпарментные модели нейрона, BRAIN в первую очередь нацелен на создание именно однокомпарментных моделей.

Симулятор CSIM — инструмент для моделирования неоднородных нейронных сетей, состоящих из точечных нейронов, в том числе - спайковых. Поддерживает моделирование как сетей, включающих всего несколько нейронов, так и крупномасштабных — до нескольких тысяч нейронов и 100000 синапсов [5]. В отличие от GENESIS и NEURON, CSIM был разработан для работы с моделями именно сетевого уровня и изучения вычислительных эффектов, которые невозможно наблюдать при моделировании единичной клетки. Дизайн CSIM, как и в NEURON и GENESIS, является объектно-ориентированным. То есть существуют объекты, связанные между собой посредством четко определенных сигнальных каналов.

Пакет моделирования Nengo ориентирован на моделирование крупномасштабных нейронных сетей — в том числе, спайковых. В отличие от традиционного нейронного моделирования Nengo не требуется правило обучения, чтобы найти весовые коэффициенты нейронной сети [6].

Симулятор NCS был разработан как максимально эффективный и простой в использовании симулятор для создания наиболее биологически правдоподобных моделей мозга [7]. Симулятор способен моделировать сеть из миллионов нейронов в режиме реального времени.

Большая часть описанных выше симуляторов позволяет описывать сегментные модели нейронов, поддерживает возможность создания пользовательских моделей, однако возможности динамического изменения параметров, а тем более топологии сети, или сегментов дендритного дерева нейрона существенно ограничены или недоступны. Тем не менее возможность описания и моделирования растущих нейронных структур (как с точки зрения изменения топологии сети нейронов, так и с точки зрения модификации дендритного дерева каждого нейрона) является актуальной задачей, решение которой позволит повысить эффективность исследования механизмов онтогенеза естественных нейронных структур и их применения в развитии искусственных нейронных сетей.

1. Архитектура программной платформы

1.1. Общее описание

Создание моделей реконфигурируемых растущих нейронных сетей подразумевает необходимость обеспечить возможность описывать системы, в которых заранее не определена как топология связей между элементами системы, так и структурное описание и топология связей составных частей элементов системы.

Целесообразно рассмотреть возможность создания не конечной среды моделирования, а унифицированной программной платформы для создания как сред моделирования, так и иных программ, требующих выполнения расчета спайковых сетей.

Реализация такой платформы может быть представлена следующими функциональными уровнями (рис.1):

1. Уровень обеспечения механизмов взаимодействия компонентов (ядро) – обеспечивающий реализацию следующих свойств:
 - а) возможность описывать сети с произвольной, в том числе нерегулярной топологией связей;
 - б) возможность описания элемента сети как объекта со сложной, динамически изменяющейся внутренней структурой;
 - с) малые накладные расходы на обмен данными между элементами сети, и их составными частями;
 - д) возможность неограниченного расширения доступных моделей и алгоритмов с произвольными требованиями к входам, выходам и параметрам;
2. Уровень управления описаниями конфигураций и расчетом, обеспечивающий
 - е) возможность создания и редактирования конфигураций, включающих в себя описаний топологии сети, параметров элементов сети и настроек расчета;
3. Уровень реализации конечных приложений расчета, обеспечивающий
 - ф) возможность создания, как приложений, выполняемых в командной строке, так и приложений с графическим интерфейсом.
4. Многопользовательская клиент-серверная среда моделирования, обеспечивающая
 - г) возможность распределенного расчета моделей на вычислительном кластере;
 - х) возможность многопользовательской работы с моделями и результатами экспериментов.

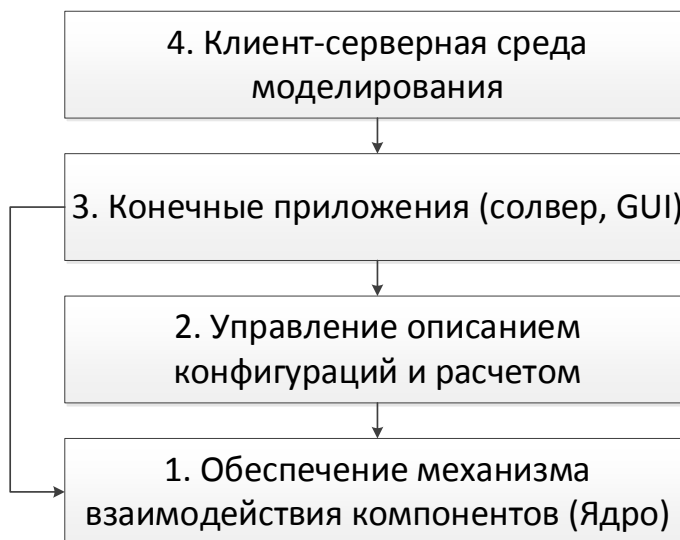


Рис.1. Функциональные уровни программной платформы

Ниже будут рассмотрены архитектурные особенности предлагаемой платформы и их вклад в решение задачи моделирования растущих спайковых нейронных сетей.

1.2. Ядро

Ядро системы (рис. 2) разрабатывается в стандарте языка ANSI C++ для обеспечения максимальной производительности и возможностей кроссплатформенной сборки. Для разработки базовых элементов пользовательского интерфейса выбран кросс-

платформенный инструмент разработки ПО Qt. Помимо этого, используя только ядро, можно создавать приложения для любых систем, поддерживающих стандарт C++ и имеющих реализацию библиотеки boost.

Ниже описаны основные составные части ядра.

Компонент – представляет собой минимальный функциональный элемент модели и содержит в себе реализацию математического описания функционального элемента моделируемого объекта. Связывание входов и выходов компонентов осуществляется во время выполнения по их символическим именам, при этом фактическое программное взаимодействие осуществляется через механизм прямого контролируемого доступа компонента-приемника данных к компоненту-источнику данных. Реализованный механизм такого доступа минимизирует накладные расходы на взаимодействие компонентов, что позволяет создавать системы, состоящие из большого числа элементов и связей. Таким образом на уровне обеспечения механизмов взаимодействия компонентов достигается возможность описания систем, состоящих из элементов, количество которых, а также число и топология связей между которыми, может изменяться во время выполнения.

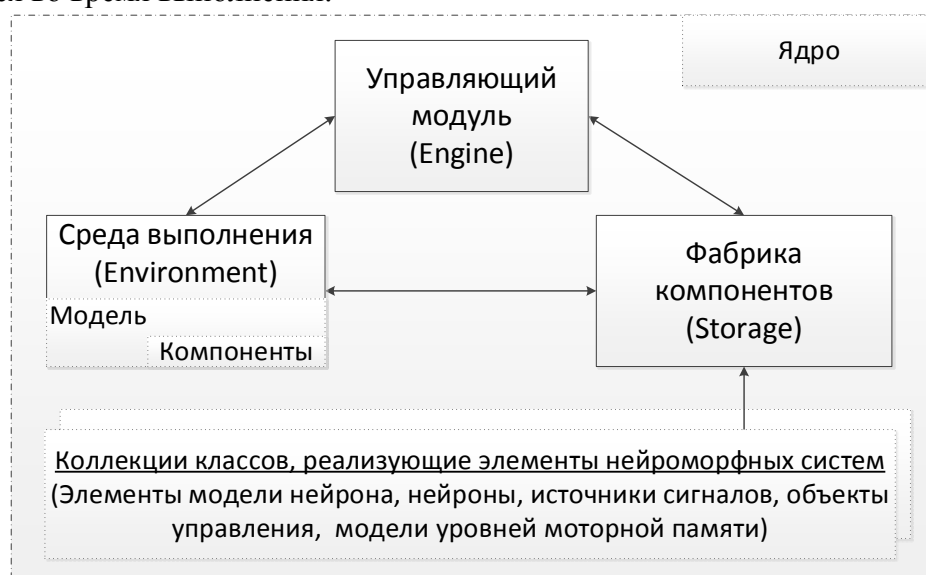


Рис. 2. Архитектура вычислительного ядра

Менеджер ядра (Manager) обеспечивает функции инициализации и деинициализации каналов расчета. Содержит обработчик исключений и главные средства журналирования, также каждый канал расчета имеет собственные средства журналирования.

Управляющий модуль (Engine) обеспечивает функционал по инициализации и деинициализации основных модулей канала расчета. Также управляющий модуль предоставляет унифицированный интерфейс взаимодействия с модулями своего канала расчета.

Фабрика компонентов (Storage) содержит коллекцию компонентов, выдает копии компонентов по требованию и следит за правильным освобождением памяти. Имплементация компонентов может находиться в динамически подключаемых библиотеках.

Среда выполнения (Environment), использует фабрику объектов для обеспечения функционала по формированию структурной схемы модели, управлению отдельными компонентами модели, заданию входных данных и параметров модели для расчета и выдачу выходных данных. Все компоненты модели обязательно должны быть определены в хранилище объектов. Каждая модель создается в отдельном канале, при этом всегда есть возможность параллельно работать с несколькими каналами, то есть с несколькими моделями. Это позволяет обрабатывать видеопоток из множества источников различными алгоритмами.

1.3. Уровни управление описанием конфигураций и расчетом и реализации конечных приложений

В эту часть входят следующие группы модулей (рис. 3): модули управления конфигурациями, модули управления расчетом и модули, обеспечивающий поддержку клиент-серверного взаимодействия.

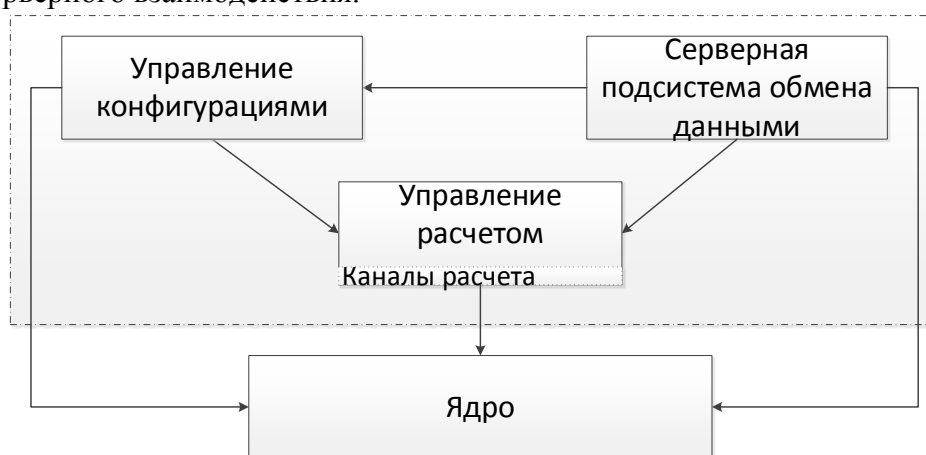


Рис. 3. Структура основных модулей уровня управления описанием конфигураций и расчетом

Уровень реализации конечных приложений (рис. 4) содержит приложение для расчета ранее созданных конфигураций, выполняемое в командной строке и кросс-платформенное приложение с графическим интерфейсом обеспечивающее возможность создания, редактирования и расчета конфигураций на локальной машине пользователя.

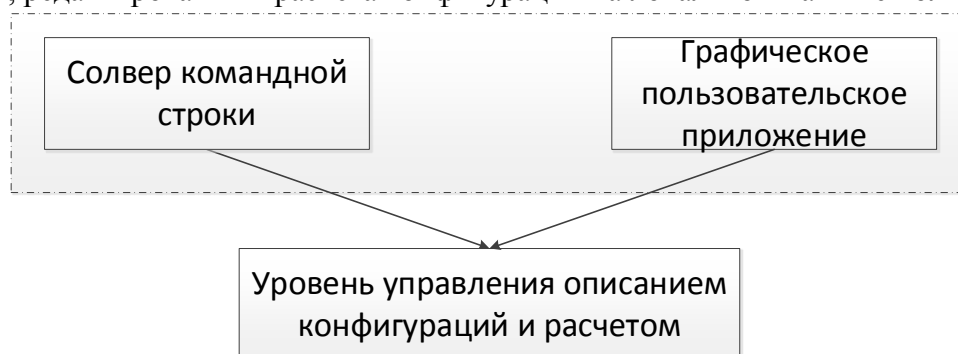


Рис. 4. Конечные приложения

1.4. Клиент-серверная среда моделирования

Уровень реализации клиент-серверной среды моделирования (рис. 5) содержит клиентское приложение, обеспечивающее функционал, аналогичный локальному

приложению пользователя, плюс возможности управления серверной подсистемой для проведения удаленных расчетов.

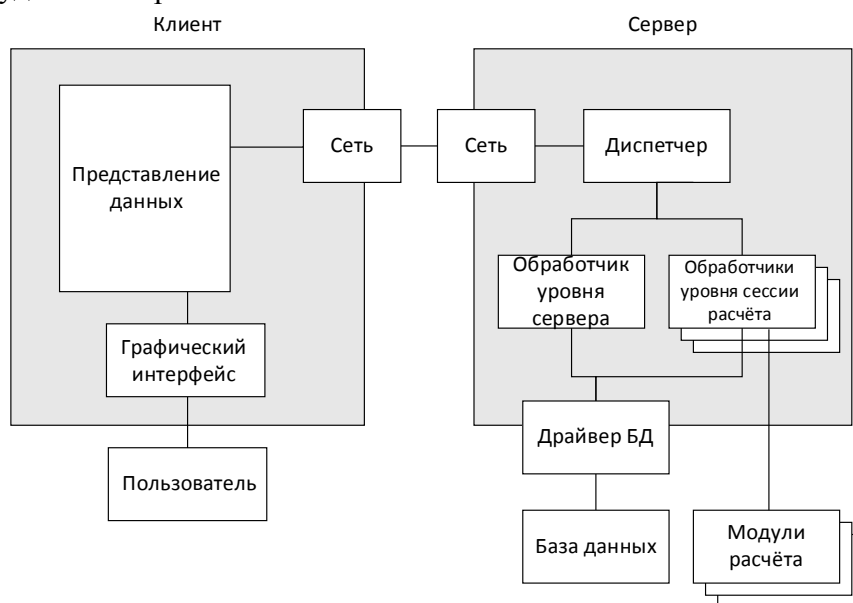


Рис. 5. Клиент-серверная среда моделирования

Сервер предоставляет пользователям сервис, имеющий достаточно большое количество вариантов использования, но при этом в пределах работы с одной сессией расчёта обращающийся к одному и тому же приложению, использующему библиотеку моделирования, что делает невозможным разнести функционал по разным модулям. Микросервисная архитектура не подходит для организации структуры сервера в целом, но возможно применить идею разделения системы на отдельные небольшие компоненты при организации работы с сессиями расчёта: так как каждая сессия расчёта работает с отдельной конфигурацией, можно запускать вычислительную часть как отдельные приложения – модули расчета (солверы), которые могут располагаться и на разных машинах.

Клиентское приложение может быть как тонким, так и толстым клиентом. Ожидается, что пользователи будут проводить длительное время за редактированием структуры моделей, состоящих из множества элементов, соединённых связями и имеющих настраиваемые параметры. Эту работу невозможно полностью автоматизировать, и она является достаточно монотонной. Так как самостоятельное десктопное приложение позволяет предоставить пользователю больше возможностей для управления приложением, включая горячие клавиши и др., то был выбран такой вариант реализации клиента.

2. Сравнение характеристик

В настоящее время авторами были реализованы уровни 1-3 разрабатываемой программной платформы (рис. 1). Платформа поддерживает создание сетей на основе следующих моделей нейронов:

- модель порогового интегратора [8];
- модель Ижикевича [9];
- авторская сегментная модель нейрона [10];

В таблице ниже приведено сравнение разрабатываемой программной платформы (Neuro Modeler) с существующими специализированными решениями без учета глубины проработки каждого решения по каждому из сравниваемых параметров.

Таблица

Сравнение характеристик сред моделирования спайковых нейронных сетей

Параметр	NEST	GENESIS	NEURON	Brian	CSIM	Nengo	NCS	Mvaspike	Neuro Modeler
<i>Возможности моделирования</i>									
Крупномасштабные сети	+	+	+	+	+	+	+	+	-
Моделирование отдельных нейронов	+	+	+	-	-	-	-	-	+
Многокомпартные модели	+	+	+	+	-	-	+	-	+
Моделирование растущих сетей	-	-	-	-	-	-	-	-	+
Поддержка пользовательских моделей	+	+	+	+	-	+	+	+	+
<i>Возможности визуализации</i>									
Наличие графического интерфейса	+	+	+	-	-	+	+	-	+
<i>Общие характеристики</i>									
Поддержка моделирования на вычислительном кластере	+	-	-	-	-	-	+	-	± (запланировано)
Кроссплатформенность	+	+	+	+	+	+	-	-	+

Можно сделать вывод о том, что разрабатываемое решение имеет по крайней мере одно конкурентное преимущество – возможность моделирования растущих нейронных сетей с заранее не определенной топологией связей.

Заключение

Основным преимуществом разрабатываемой программной платформы является возможность моделирования растущих спайковых нейронных сетей с заранее не определенной топологией связей, в которых каждый функциональный элемент – нейрон может быть описан сегментной моделью с изменяемой структурой дендритного дерева.

Недостатком среды является принципиальная сложность обеспечения параллелизации вычислений моделей нейронов и их сегментов, что, однако может быть частично преодолено за счет разделения моделируемой сети на отдельные подсистемы, расчет каждой из которых выполняется независимо отдельным модулем.

Применение разрабатываемой платформы может быть ограничено на любом из описанных уровней в зависимости от текущих требований. При этом целесообразно ограничиться уровнем 1 при разработке библиотек, использующих внешнюю подсистему расчета (например интеграция с другими средами моделирования, такими как Matlab, или создание встраиваемых систем); уровни 2-3 целесообразно использовать при разработке специализированных программных средств, включая локально выполняемые среды моделирования; уровень 4 при создании облачной многопользовательской системы.

Список литературы

1. NEST: The Neural Simulation Tool: официальный сайт. — URL: <http://www.nest-simulator.org/> (дата обращения: 20.07.2020) – Режим доступа: свободный. — Текст: электронный.
2. GENESIS: The GEneral NEural SIMulation System: официальный сайт. — URL: <http://www.genesis-sim.org/> (дата обращения: 20.07.2020). — Режим доступа: свободный. — Текст: электронный.
3. Brette R. Simulation of networks of spiking neurons: A review of tools and strategies / R. Brette, M. Rudolph, T. Carnevale, M. Hines, D. Beeman, J. M. Bower, M. Diesmann, A. I. Morrison, P. H. Goodman, F. C. Harris Jr., M. Zirpe, T. Natschläger, D. Pecevski, B. Ermentrout, M. Djurfeldt, A. Lansner, O. Rochel, T. Vieville, E. Muller, A. P. Davison, S. El Boustani, A. Destexhe. — DOI 10.1007/s10827-007-0038-6 // J Comput Neurosci. — 2007. — № 23(3). — С. 349–398. Доступен по адресу: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2638500/> (дата обращения: 20.07.2020).
4. Brian. The Brian spiking neural network simulator: официальный сайт. — URL: <http://briansimulator.org/> (дата обращения: 20.07.2020). — Режим доступа: свободный. — Текст: электронный.
5. CSIM. A neural Circuit SIMulator: официальный сайт. — URL: <http://www.lsm.tugraz.at/csim/index.html> (дата обращения: 20.07.2020). — Режим доступа: свободный. — Текст: электронный.
6. Nengo // The Nengo Neural Simulator: official website. Available at: <http://nengo.ca/>, accessed 06.10.2015.
7. Brain Computation Lab: официальный сайт. — URL: <https://www.cse.unr.edu/brain/ncs> (дата обращения: 20.07.2020). — Режим доступа: свободный. — Текст: электронный.
8. Brette R and Gerstner W (2005). Adaptive Exponential Integrate-and-Fire Model as an Effective Description of Neuronal Activity. J Neurophysiol 94:3637-3642 DOI: <https://doi.org/10.1152/jn.00686.2005>
9. Izhikevich, E.M. Simple model of spiking neurons / E.M. Izhikevich // IEEE transactions on neural networks: a publication of the IEEE Neural Networks Council. – 2003. – Vol. 14, № 6. – P. 1569-1572.
10. Bakhshiev A.V. Mathematical Model of the Impulses Transformation Processes in Natural Neurons for Biologically Inspired Control Systems Development / A.V. Bakhshiev, F.V. Gundelakh // Supplementary Proceedings of the 4th International Conference on Analysis of Images, Social Networks and Texts (AIST-SUP 2015), Yekaterinburg, Russia, April 9-11, 2015. Published on CEUR-WS, vol. 1452, pp. 1-12, 15.10.2015. Online: <http://ceur-ws.org/Vol-1452/>

ОПРЕДЕЛЕНИЕ ТЕМПЕРАТУРЫ МОЗГА ПО МАГНИТНО-РЕЗОНАНСНЫМ ТОМОГРАФИЧЕСКИМ ИЗОБРАЖЕНИЯМ

Н.М. Богатов, Л.Р. Григорьян, А.С. Сеницын, С.К. Шаповалов
Кубанский государственный университет, bogatov@phys.kubsu.ru

Введение

При нейротравмах, после субарахноидального кровоизлияния или инсульта возникает температурная гетерогенность мозга с очагами высокой температуры [1, 2]. Температура головного мозга зависит от метаболической активности нейронов и, следовательно, влияет на функции центральной нервной системы и обменные процессы в организме [1]. Разработка методов неинвазивной термометрии головного мозга является актуальной задачей. Такими методами являются магниторезонансная термометрия [3, 4], СВЧ-радиотермометрия [5, 6], акустотермометрия [7, 8]. Сравнение этих методов выполнено в [9], показано, что наилучшее пространственное разрешение имеет магниторезонансная термометрия.

Разнообразие методов построения тепловых изображений с помощью МРТ обусловлено многофакторностью влияния температуры на состояние живого объекта, что выражается в температурной зависимости измеряемых параметров: времени продольной (спин-решеточной) релаксации T_1 , времени поперечной (спин-спиновой) релаксации T_2 , намагниченности M_0 , связанной с протонной плотностью, коэффициента диффузии D , химического сдвига δ . Сравнительный анализ различных методов проведен в работе [4].

Регистрация теплового поля методом МРТ в объеме для различных способов нагревания заданной области обсуждается в следующих работах: микроволновой нагрев [10], лазерный нагрев [11], нагрев от миниатюрных электронагревателей [12].

Времена релаксации T_1 , T_2 зависят от индукции магнитного поля, эта зависимость для T_1 более выражена, чем для T_2 [13]. В результате контрастные характеристики биологической ткани на T_1 МРТ изображениях зависят от значения индукции магнитного поля [14].

В настоящее время развиваются методы целенаправленного нагревания тканей организма с целью активизации или подавления химико-биологических процессов: местная (локальная), региональная и гипертермия всего тела. Местная гипертермия требует высокой степени контроля воздействий на ткань, при этом используются ультразвуковые, лазерные, РЧ, СВЧ воздействия. Возможность точной магнитно-резонансной (МР) термометрии с коррекцией движения в клинической практике полной абляции опухоли печени показана в работе [15].

Важность калибровочных измерений в МР термометрии для оценки температуры головного мозга человека в клинических исследованиях продемонстрирована в работе [16]. Физические принципы, лежащие в основе медико-биологических применений МР визуализации температуры, и нерешенные проблемы рассмотрены в обзорной статье [17].

Целью данной работы является определение температуры мозга на основе цифрового анализа T_2 – взвешенных МРТ изображений.

Анализировалось изменение интенсивности I серого цвета пикселей МР изображений, взвешенных по времени спин-спиновой релаксации протонов T_2 . Значения интенсивности пикселей I в исходных изображениях нормированы так, что $I \in [0, 255]$. Повышение разрешающей способности достигалось использованием псевдоцветовой палитры МР изображений с помощью программы [18] и методов преобразования изображений [19]. Поверхностная температура исследуемых фантомов, однородных и неоднородных по составу, измерялась с помощью контактного термодатчика Nihon Konden BSN 2301K.

Цифровой анализ МРТ изображений фантомов

Фантомы, неоднородные по составу, исследовались на магнитно-резонансном томографе 1T Raportana HFO 1.0. Фантомы D, E – пластиковые бутылки 1,5 л, заполненные каждая водой (D1, E1) и рафинированным подсолнечным маслом (D2, E2). Фантом D имел температуру $t_D = 13^\circ\text{C}$, фантом E – $t_E = 38^\circ\text{C}$. На рис. 1 показано T_2 – взвешенное изображение аксиального сечения фантомов D, E, полученное с помощью импульсной последовательности TSE. На рис. 2, рис. 3 приведена зависимость интенсивности I от длины L , измеряемой в пикселях, для линий 1, 2, 3, 4 (рис. 1), соответственно.

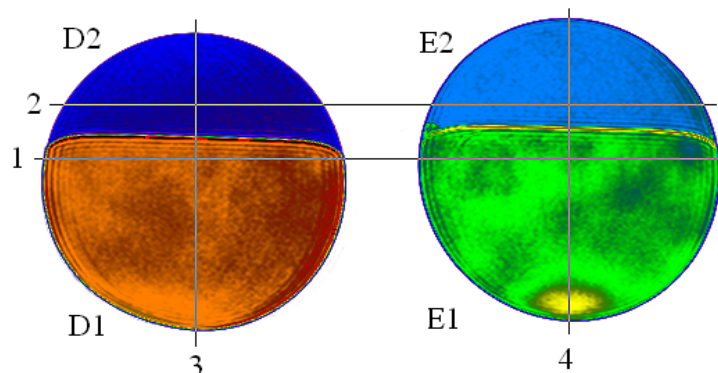


Рис.1. T_2 – взвешенное изображение неоднородных фантомов D, E: D1 – вода; D2 – подсолнечное масло; E1 – вода; E2 – подсолнечное масло.

T_2 – взвешенное изображение фантомов D, E в областях D1, D2, E1, E2 достаточно однородно (рис. 1), что подтверждается распределением интенсивности сигнала I вдоль линий (рис. 2, рис. 3).

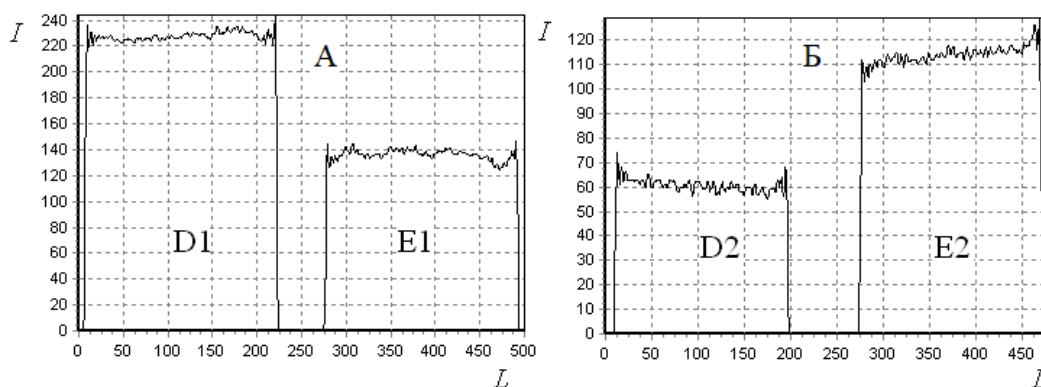


Рис.2. Распределение интенсивности сигнала I :
А – вдоль линии 1 рис. 1, Б – вдоль линии 2 рис. 1

Интенсивность I на T_2 –взвешенном изображении уменьшается с ростом температуры для воды (рис. 2А), но увеличивается для подсолнечного масла (рис. 2Б).

Разность значений ΔI на границе масло-вода T_2 –взвешенного изображения демонстрирует следующую закономерность: для фантома D (рис. 3А)

$$\Delta I|_D = I|_{D2} - I|_{D1} \approx -170,$$

это значение по модулю больше, чем для фантома Е (рис. 3Б)

$$\Delta I|_E = I|_{E2} - I|_{E1} \approx -25.$$

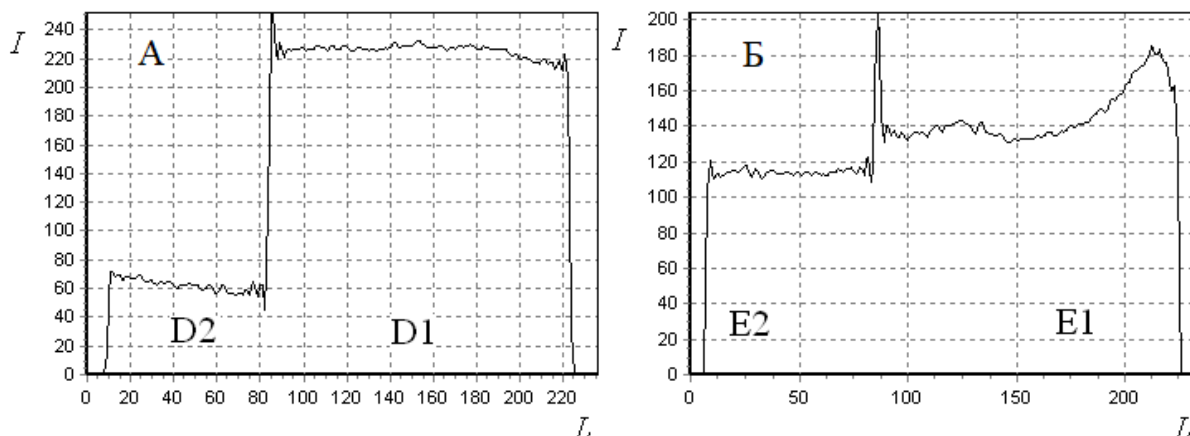


Рис.3. Распределение интенсивности сигнала I :
А – вдоль линии 3 рис. 1, Б – вдоль линии 4 рис. 1

Для фантома Е на границе масло-вода со стороны воды наблюдается резкий пик сигнала T_2 (рис. 3Б) – химический сдвиг. Значение химического сдвига δ для фантома Е значительно больше, чем для фантома D (рис. 3А), что свидетельствует о необходимости изучения зависимости $\delta(t)$ с целью использования ее для определения температуры.

Увеличение интенсивности I в области E_1 (рис. 3Б) в направлении внешней границы требует дальнейшего исследования. Возможно, оно связано с диффузией более легких атомов из области E_2 в область E_1 в фантоме с более высокой температурой.

Среднее значение интенсивности сигнала I для веществ D1, E1 и D2, E2, составляющих фантомы D, Е, рассчитанные во внутренних прямоугольных областях: $I_{D1} = 229,2$; $I_{E1} = 137,5$; $I_{D2} = 60,8$; $I_{E2} = 113,5$. Стандартные отклонения, соответствующие этим величинам: $S_{D1} = 3,3$; $S_{E1} = 4,5$; $S_{D2} = 2,5$; $S_{E2} = 2,1$. Интервалы $I_{D1} \pm S_{D1}$; $I_{E1} \pm S_{E1}$; $I_{D2} \pm S_{D2}$; $I_{E2} \pm S_{E2}$ не перекрываются. Следовательно, температурное состояние веществ D1, E1, D2, E2 определяется однозначно по T_2 –взвешенному изображению.

Определение температуры мозга

Исследования выполнялись на магнитно-резонансном томографе 1,5T Toshiba. Фантом, однородный по составу, был составлен из 5 пластиковых банок объемом 100 мл, заполненных водой с различной температурой: 1 – $t_1 = 15$ °C, 2 – $t_2 = 21$ °C, 3 – $t_3 = 33$ °C, 4 – $t_4 = 38$ °C, 5 – $t_5 = 50$ °C. T_2 –взвешенные изображения в аксиальных сечениях фантома получены с помощью импульсной последовательности TSE. На основе этих изображений составлена таблица соответствия температуры и параметров интенсивности серого цвета в областях с заданной температурой: I_{av} – среднего значения,

I_{min} – минимального значения, I_{max} – максимального значения, S – стандартного отклонения, усредненных по нескольким срезам.

По значениям $I_{av}(t_i)$, где $i = 1, 2, 3, 4, 5$, построена линия тренда (рис. 4), коэффициенты которой использованы в формуле для вычисления температуры жидкости:

$$t = (194,6 - I_{av}) / 1,846. \quad (1)$$

На рис. 5 представлено T_2 –взвешенное изображение аксиального сечения головы в цветной кодировке. С правой стороны от головы находится тест-объект – пластиковая бутылка с водой. На цветовой палитре отложены значения интенсивности I цвета исходного серого изображения.

Таблица
Параметры интенсивности серого цвета

№	$t, ^\circ\text{C}$	I_{av}	I_{min}	I_{max}	S
1	15	162	151	200	4,9
2	21	161	146	195	6,8
3	33	131	123	142	2,9
4	38	131	121	148	3,7
5	50	98	93	108	2,4

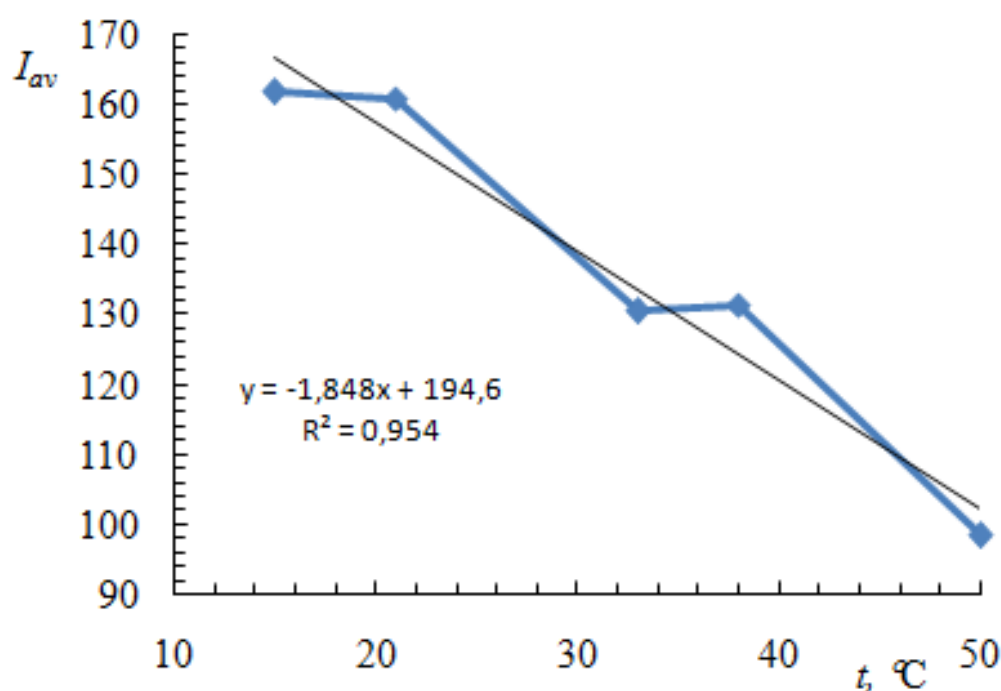


Рис.4. Зависимость интенсивности цвета от температуры для аксиальных T_2 –взвешенных изображений фантома.

Наименьшую температуру на рис. 5 имеет тест-объект. Поэтому вода тест-объекта выглядит наиболее яркой. Средняя интенсивность цвета для тест-объекта I_{av}

= 140,3, подставив это значение в формулу (1), получим среднюю температуру воды $t = 29,3$ °C.

Яркость спинномозговой жидкости оказывается меньше, так как с увеличением температуры яркость изображения жидкости уменьшается. Средняя интенсивность цвета боковых желудочков $I_{av} = 125,2$, подставив это значение в формулу (1), получим среднюю температуру спинномозговой жидкости $t = 37,6$ °C, что согласуется с нормальным физиологическим значением температуры мозга.

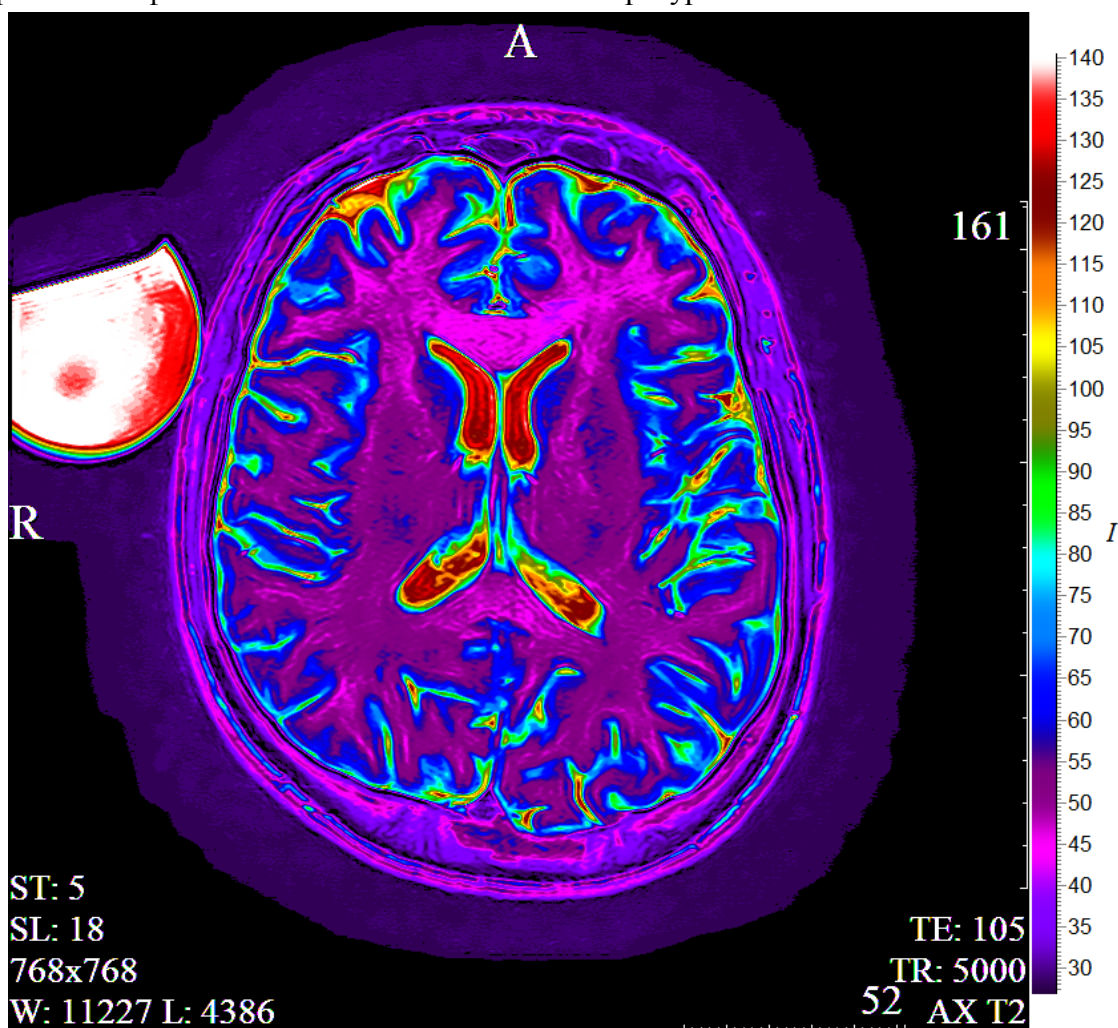


Рис.5. T_2 –взвешенное изображение головы и тест-объекта.

Обсуждение результатов

T_2 –взвешенные изображения однородных фантомов, полученные с помощью импульсной последовательности Turbo Spin Echo, имеют достаточно однородное распределение интенсивности цвета, поэтому их целесообразно использовать для идентификации температурного состояния объектов. Однако в этом случае возрастают требования к точности модели, описывающей зависимость времени спин-спиновой релаксации T_2 от индукции магнитного поля, температуры и химического состава исследуемого вещества.

Для объектов неоднородного состава на T_2 изображениях ярко проявляется температурная зависимость химического сдвига на границе раздела сред. Кроме того, при построении модели следует учитывать влияние температуры на диффузию молекул в область с их меньшей концентрацией.

Показано, что интенсивность цвета на T_2 –взвешенном изображении фантома, содержащего воду с различной температурой, полученном с помощью импульсной последовательности TSE, уменьшается с ростом температуры воды. Используя эту зависимость как калибровочную при анализе T_2 –взвешенного изображения головы, полученного с помощью последовательности TSE, рассчитана температура спинномозговой жидкости. Найденное значение соответствует физиологическому значению температуры мозга.

Список литературы

1. Mrozek S. Brain Temperature: Physiology and Pathophysiology after Brain Injury / S. Mrozek, S. Vardon S., T. Geeraerts // *Anesthesiology Research and Practice*. — 2012. — ID 989487. — P.1 – 13.
2. Idris Z. Neurointensive Care Monitoring for Severe Traumatic Brain Injury / Z. Idris, M. Mustapha, J. M. Abdullah // *Brain Injury - Pathogenesis, Monitoring, Recovery and Management*. — Ed. Prof. Amit Agrawal. — 2012. — P.11 – 244.
3. Гуляев М.В. Магнитно-резонансная термометрия головного мозга крыс методом локальной ЯМР спектроскопии / М.В. Гуляев, С.К. Ханов, Д.Д. Наместникова и др. // *Журнал Радиоэлектроники*. — 2013. — №10. — С.1 – 10. E-print: <http://jre.cplire.ru/jre/oct13/11/text.pdf>
4. Lüdemann L. Wust Non-invasive magnetic resonance thermography during regional hyperthermia / L. Lüdemann, W. Wlodarczyk, J. Nadobny, et al. // *Int. J. Hyperthermia*. — 2010. — V.26. — I.3. — P.273 – 282.
5. Чебоксаров Д.В. Диагностические возможности неинвазивного термомониторинга головного мозга / Д.В. Чебоксаров, А.В. Бутров, О.А. Шевелев и др. / *Анестезиология и реаниматология*. — 2015. — Т.60. — №1. — С.66 – 69.
6. Шевелев О.А. Патогенетическая роль церебральной гипертермии при поражениях головного мозга / О.А. Шевелев, А.В. Бутров, Д.В. Чебоксаров и др. // *Клиническая медицина*. — 2017. — Т.95. — №4. — С.302 – 309.
7. Аносов А.А. Использование априорной информации в решении динамических обратных задач акустотермометрии / А.А. Аносов // *Журнал Радиоэлектроники*. — 2016. — №6. — С.1 – 24. E-print: <http://jre.cplire.ru/jre/jun16/8/text.pdf>
8. Найда С.А. Фокусируемые пьезоприемники для акустотермометрии / С.А. Найда, Е.С. Дрозденко // *Техническая диагностика и неразрушающий контроль*. — 2013. — №3. — С.41 – 47.
9. Жорина Л.В. Методы неинвазивного измерения внутренней температуры тела / Л.В. Жорина // *Вестник Тамбовского университета. Серия Естественные и технические науки*. — 2017. — Т.22. — В.2. — С. 464 – 470.

10. Vitkin I.A. Magnetic resonance imaging of temperature changes during interstitial microwave heating: a phantom study / I.A. Vitkin, J.A. Moriarty, R.D. Peters, et al. // *Med. Phys.* — 1997. — V.24. — I.2. — P.269 – 277.
11. Волков А.А. Регистрация температурного поля методом магнитно-резонансной томографии / А.А. Волков, В.Н. Никифоров, Ю.А. Пирогов и др. // *Медицинская физика.* — 2011. — №1. — С.75 – 81.
12. Волков А.А. Традиционные методы нагрева в применении к магнитно-резонансной термометрии / А.А. Волков, С.К. Какагельдыев, А.С. Прохоров и др. // *Журнал Радиоэлектроники.* — 2012. — №1. — С.1 – 18. E-print: <http://jre.cplire.ru/iso/jan12/12/text.html>
13. Богатов Н.М. Влияние температуры объекта на магнитно-резонансные томографические изображения / Н.М. Богатов, С.А. Сеницын // *Современные проблемы физики, биофизики и инфокоммуникационных технологий.* Краснодар: ЦНТИ. — 2013. — С.91–105.
14. Богатов Н.М. Влияние градиентного магнитного поля на магнитно-резонансные томографические изображения фантомов с различной температурой / Н.М. Богатов, С.А. Сеницын // *Интернет-журнал «Наукоедение».* — 2014. — №4 (23). — С.1 – 17. E-print: <http://naukovedenie.ru/PDF/120TVN414.pdf>
15. Kägebein U. Motion Correction in Proton Resonance Frequency-based Thermometry in the Liver / U. Kägebein, O. Speck, F. Wacker et al. // *Topics in Magnetic Resonance Imaging.* — 2018. — V.27. — N.1. — P. 53 – 61.
16. Verius M. Magnetic Resonance Spectroscopy Thermometry at 3 Tesla: Importance of Calibration Measurements / M. Verius, F. Frank, E. Gizewski et al. // *Therapeutic Hypothermia and Temperature Management.* — 2019. — V.9. — N.2. — P.146 – 155.
17. Odéen H. Magnetic Resonance Thermometry and Its Biological Applications - Physical Principles and Practical Considerations / H. Odéen, D.L. Parker // *Progress in Nuclear Magnetic Resonance Spectroscopy.* — 2019. — V.110. — P.34 – 61.
18. Свидетельство о государственной регистрации программы для ЭВМ. Российская Федерация. Программа преобразования псевдоцветовых изображений / Н.М. Богатов, Л.Р. Григорьян, М.С. Коваленко и др. — №2017611900. Зарегистрировано в Реестре программ для ЭВМ: 10.02.2017 г.
19. Богатов Н.М. Преобразование цветовой палитры изображений магнитно-резонансной томографии / Н.М. Богатов, Л.Р. Григорьян, А.С. Сеницын // *Современные проблемы физики, биофизики и инфокоммуникационных технологий.* Краснодар: Краснодарский ЦНТИ. — 2012. — С.45 – 60.

ИДЕНТИФИКАЦИЯ МОДЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ АНАЛОГОВ ФУНКЦИЙ ЧУВСТВИТЕЛЬНОСТИ

В.С.Воронов¹, А.И.Рубан²

Сибирский федеральный университет,

Институт космических и информационных технологий, каф. информатики

¹*voronov.volodya2013@yandex.ru*

²*ai-rouban@mail.ru*

За последнее десятилетие интерес к анализу чувствительности (АЧ) продолжает возрастать. Анализ чувствительности получает активное применение в смежных науках, таких как физика, химия, биология медицина и др. Это связано с возрастающей сложностью моделей за счет расширения учитываемых факторов, наличия неопределенности и др. и, как следствие, увеличения количества входных параметров. Анализ чувствительности позволяет не только ранжировать параметры модели по степени влияния на выходные значения, но и проводить структурную оптимизацию модели [1]. Методы анализа чувствительности можно разделить на две категории: локальные и глобальные методы. Наибольшей популярностью пользуется локальный анализ чувствительности. Он проводится относительно набора номинальных параметров. Несмотря на простоту локальные методы вычисления функций чувствительности (ФЧ) обладают некоторыми недостатками. Один из них малая область варьирования параметров. Устранить недостатки локального АЧ позволяет глобальный анализ чувствительности. Он хорошо описан в литературе, но используется гораздо реже [2, 3]. Практика АЧ применительно к идентификации помогает упростить модель, позволяя отбросить из рассмотрения незначительные параметры, или использоваться непосредственно для их вычисления в градиентных методах например, в методах на основе стохастического градиента или методах наименьших квадратов. Основная сложность в вычислении ФЧ состоит в необходимости аналитического составления уравнения чувствительности, количество которых совпадает с количеством параметров, с последующим их решением. К сожалению, это бывает затруднительно в силу ряда причин, таких как большое количество уравнений чувствительности, сложность или недоступность модели. Это привело к появлению методов численной оценки как локальной [4, 5], так и глобальной [1, 6] чувствительности. Применение конкретного метода зависит от задачи [3], поэтому разработка новых методов остается актуальной проблемой. Стоит отметить, что для адаптивной идентификации, где сбор информации происходит постепенно, необходимо адаптировать большинство методов оценки ФЧ, т.к. они основаны на ретроспективной обработке измерений.

Пусть поведение объекта в динамическом режиме описывается разностным уравнением

$$x(t) = f(x(t-1), u(t-\tau-1), a)$$

где τ – величина чистого запаздывания, $\tau \in Z_{\geq 0}$, a – истинные параметры объекта, $a \in R^d$. На основе этого представления строиться модель объекта с переменными параметрами

$$y(t | \alpha(t)) = f(x(t-1), u(t-\tau-1), \alpha(t)) = \nabla_{\alpha}^T y(t | \alpha(t)) \alpha(t). \quad (1)$$

Использование функций чувствительности в алгоритмах идентификации основано на линеаризации модели [7] $y(t | \alpha(t)) \approx y(t | \alpha(t-1)) + S^T(t) \Delta \alpha(t)$, где $S(t)$ – вектор-столбец функций чувствительности, $S(t) \in R^d$.

Необходимо по доступным к моменту времени t измерениям входов и выходов объекта $\{\{x(i), u(i)\}, i = \overline{1, N}\}$, где N – размер выборки, оценить вектор параметров $\alpha(t) \in R^d$, используя численный метод вычисления функций чувствительности.

Предлагаемый алгоритм основывается на использовании локального метода анализа чувствительности при одновременном варьировании параметров в широкой параметрической области, что характерно для глобальных методов. В качестве локального метода используется метод конечных разностей (МКН). Понятно, что вычисление ФЧ с помощью конечных разностей в области больших размеров не гарантирует получение точной оценки. Как отмечается в [5] это является основным недостатком МКН. Получаемые таким образом функции здесь именуются аналогами ФЧ. Стоит отметить, что при уменьшении размеров параметрической области ε аналоги ФЧ сходятся к их оценкам, т.е. $\lim_{\varepsilon \rightarrow 0} S_\varepsilon = \frac{\partial y}{\partial \alpha}$, где S_ε – аналог ФЧ.

Ретроспективный итеративный метод идентификации с применением аналогов функций чувствительности состоит из трех этапов. На первом этапе происходит вычисление аналогов ФЧ. Этот этап основан на применении метода конечных разностей с одновременным варьированием всех параметров в широкой области. После чего вычисляются оценки параметров с помощью любого градиентного метода. Далее идет сужение области анализа. Эти шаги повторяются до выполнения условия останова.

Рассмотрим n вариаций параметров относительно базовой точки $\bar{\alpha}_i$ в области варьирования параметров размером ε_i

$$\alpha_{i,j} = \bar{\alpha}_i + \varepsilon_i u_{i,j} = \bar{\alpha}_i + \Delta \alpha_{i,j}, \quad u_{i,j} \sim U(-1,1), \quad j = \overline{1, n}. \quad (2)$$

Здесь U – равномерный закон распределения. В качестве инструмента для размещения точек можно использовать не только генератор псевдослучайных чисел, но и последовательности с минимальным расхождением (например, R-квадрат) или оптимальные планы.

Основываясь на методе конечных разностей, получаем систему из n линейных уравнений

$$\Delta \alpha_i S_i = \Delta y(t | \alpha_i), \quad \Delta y(t | \alpha_{i,j}) = y(t | \bar{\alpha}_i) - y(t | \alpha_{i,j}), \quad \Delta y(t | \alpha_i) = (\Delta y(t | \alpha_{i,j})), \quad (3)$$

где матрица $\Delta \alpha_i = (\Delta \alpha_{i,j}^T)^T \in R^{n \times d}$, $S_i = (S_{i,j}(t)) \in R^{d \times N}$, а $\Delta y(t | \alpha_i) \in R^{n \times N}$, $j = \overline{1, n}$, $t = \overline{1, N}$. Из уравнения (3) находим S_i , например, по методу наименьших квадратов (МНК).

Второй этап алгоритма заключается в оценке неизвестных параметров модели (1). Используя ретроспективный МНК

$$S_i S_i^T \Delta \bar{\alpha}_{i+1} = S_i (x - y(\bar{\alpha}_i)), \quad x = (x(t))^T, \quad y(\bar{\alpha}_i) = (y(\bar{\alpha}_i))^T, \quad t = \overline{1, N}, \quad (4)$$

находим приращения параметров $\Delta \bar{\alpha}_{i+1}$. После этого находим параметры модели $\bar{\alpha}_{i+1} = \bar{\alpha}_i + \gamma_i \Delta \bar{\alpha}_{i+1}$. Коэффициент $\gamma_i > 0$ выбирается исходя из условия монотонной сходимости квадратичного критерия, и влияет на скорость сходимости алгоритма. Например, в качестве значений γ_i можно использовать элементы убывающей последовательности $(1/2^j)_{j=0}^{\infty}$.

Изменение размеров области поиска основано на подходе, использованном в методе селективного усреднения координат [8],

$$\varepsilon_{i+1} = \lambda \varepsilon_i \left(\sum_{j=1}^n |u_{i,j}|^q \bar{p}_{i,j}^{-(s)} \right)^{\frac{1}{q}}, \quad \bar{p}_{i,j}^{(s)} = \frac{p_s(g_{i,j})}{\sum_{k=1}^n p_s(g_{i,k})}, \quad g_{i,j} = \frac{J(\alpha_{i,j}) - J_j^{\min}}{J_j^{\max} - J_j^{\min}}, \quad (5)$$

$$J_j^{\min} = \min\{J(\alpha_{i,j}), j = \overline{1, n}\}, \quad J_j^{\max} = \max\{J(\alpha_{i,j}), j = \overline{1, n}\}, \quad J(\alpha) = \sum_{t=1}^N (x(t) - y(t | \alpha)).$$

Здесь $q \geq 1$, p_s – ядерная функция, варианты и свойства некоторых из них описаны в [8]. Практика показывает, что достаточно использовать параболические ядра $p_s(g) = (1 - g^2)^s$. Коэффициент $\lambda > 0$ задает скорость стягивание области вокруг рабочей точки $\bar{\alpha}_i$. Рекомендуется выбирать значение из интервала $[0.8, 1.2]$.

Адаптивный метод идентификации можно получить из (4) аналогично получению адаптивного МНК. Рекуррентные соотношения будут следующими

$$\gamma_{i,j} = (A_{S_i,j-1}^{-1} \Delta \alpha_{i,j}^T) (1 + \Delta \alpha_{i,j} A_{S_i,j-1}^{-1} \Delta \alpha_{i,j}^T)^{-1} = (A_{S_i,j-1}^{-1} \Delta \alpha_{i,j}^T) B_{S_i,j}^{-1}, \quad (6)$$

$$S_{i,j} = S_{i,j-1} + \gamma_{i,j} (\Delta y_{i,j} - \Delta \alpha_{i,j} S_{i,j-1}), \quad S_{i,0} = S_{i-1}, \quad (7)$$

$$A_{S_i,j}^{-1} = (E - \gamma_{i,j} \Delta \alpha_{i,j}) A_{S_i,j-1}^{-1}, \quad A_{S_i,0}^{-1} = (\Delta \alpha_0^T \Delta \alpha_0)^{-1}, \quad i = 1, 2, \dots, \quad j = 1, 2, \dots \quad (8)$$

Структура алгоритма (6) схожа со структурой адаптивного МНК [8]. Различия заключаются в матрице B , и матрице A , которая состоит из приращений параметров относительно рабочей точки. Начальные значения S_0 задаются исследователем. Алгоритм (6) – (8) помимо основных итераций по измерениям (i) входов и выходов имеет «внутренних» итерации (j) для вычисления ФЧ. Адаптивный алгоритм повторяет шаги алгоритма (2) – (5). В начале каждого цикла внутренних итераций ($j = 0$) происходит генерация точек в области размером ε_i относительно $\bar{\alpha}_i$ как в (2). Вычисляются вектор аналогов ФЧ $S_{i,j}$. Затем происходит изменение размеров области поиска. В качестве процедуры изменения размеров можно использовать (5) или более простую $\varepsilon_i = \beta_i \varepsilon_{i-1}$, где $\beta_i \in (0, 1)$. При $\varepsilon_i \rightarrow 0$ аналоги ФЧ $S_{i,j}$ стремятся к точной их оценке. Нетрудно показать, что с уменьшением области варьирования параметров матрица B_{S_N} будет сходиться к матрице единиц. Таким образом, критерием останова внутренней итеративной процедуры можно принять условие $|B_{S_i,j}| = 0$ в случае, если $d > 1$, и $|B_{S_i,j}| = 1$, если $d = 1$.

Для иллюстрации эффективности работы алгоритма используются две разные модели. Первый простой пример из литературы представлен в качестве модели Моно, описывающей рост микроорганизмов

$$\dot{s} = -\frac{1}{y} \frac{\mu_{\max} s}{k_s + s} x, \quad (9)$$

$$\dot{x} = \frac{\mu_{\max} s}{k_s + s} x - k_d x. \quad (10)$$

Здесь s – концентрация ограничивающего субстрата для роста, x – биомасса, $\mu_{\max}$ – максимальная скорость роста, y – коэффициент урожайности, k_s – «константа половинной скорости», k_d – коэффициент распада. Начальные условия для уравнений (9) и (10) примем аналогично [5], $s_0 = 10$, $x_0 = 0.1$, $\mu_{\max} = 0.5$, $y = 0.5$, $k_s = 20$, $k_d = 0.03$. Чувствительность подобной модели была изучена с помощью прямого дифференцирования в [9] и с помощью метода конечных разностей в [5]. ФЧ S_μ , S_{k_s} и S_y , полученные с помощью алгоритма (6) – (8), приведены на Рис.1.

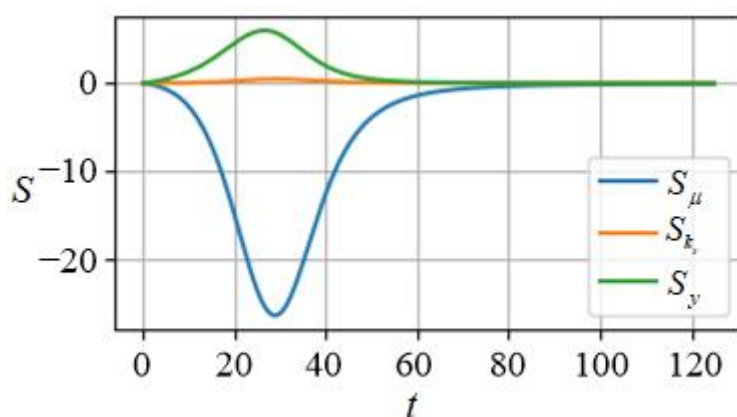


Рис.1. Чувствительность модели (9) относительно параметров $\mu_{\max}$, k_s и y .

Второй моделью была выбрана линейная относительно параметров динамическая модель процесса простой дистилляции. Процесс протекает в дистилляторе и предназначен для первичной очистки жидкостей. Этот процесс известен как медленный, вследствие значительной тепловой инерции системы и наличия чистого запаздывания. В связи с конструктивными особенностями объекта интерес представляет динамика нагрева жидкости, при этом охлаждение жидкости осуществляется оттоком тепла в окружающую среду. Зачастую фундаментальная динамика процессов нагрева описывается моделями первого и второго порядка. Модель процесса можно представить в виде разностного уравнения с чистым запаздыванием

$$y(t | \alpha) = \alpha_0 + \alpha_1 x(t-1) + \alpha_2 x(t-2) + \alpha_3 x(t-3) + \alpha_4 u(t-3) + \alpha_5 u(t-4) \quad (11)$$

с начальными условиями $x(0) = 27.4$, $u(0) = 0$.

Для идентификации были получены данные с помощью отклика на ступенчатое воздействие при $u(0) = 100$ и $t \in [0, 200]$.

Алгоритм (2) – (5) дает модель процесса с параметрами: $\alpha_0 = 6.1 \cdot 10^{-5}$, $\alpha_1 = 0.975$, $\alpha_2 = 1$, $\alpha_3 = -0.975$, $\alpha_4 = 2.56 \cdot 10^{-4}$, $\alpha_5 = 6.62 \cdot 10^{-5}$.

Качество идентификации оценивается по критерию среднеквадратичной ошибки $MSE = \frac{1}{N} \sum_{i=1}^N (x(i) - y(i|\alpha))^2$. Для приведенного результата $MSE = 1.94 \cdot 10^{-6}$. Качество идентификации слабо зависит от входных параметров изменения области поиска λ и s . Рекомендуется использовать $\lambda = 1$, а $s = 5$ для параболического ядра. При $n \geq 10$ $MSE \approx 0$ и существенно не изменяется.

Адаптивный алгоритм (6) – (8) протестирован с дальнейшей идентификацией с помощью адаптивного МНК. Предлагаемый метод не уступает по качеству работы ретроспективному варианту. При использовании $n = 5$, простого варианта сжатия области $\varepsilon_i = 0.5\varepsilon_{i-1}$ и остальных параметрах аналогичных ретроспективному методу алгоритм (6) – (8) имеет $MSE_{S_0} = 5.95 \cdot 10^{-10}$ и $MSE_{S_1} = MSE_{S_4} = 0$. Показатели были вычислены относительно ФЧ, полученных аналитически. Стоит отметить, что скорость сходимости параметров для модели (11), вычисляемых с помощью адаптивного МНК (Рис.2), не отличается от характеристик, получаемых при аналитическом вычислении ФЧ. Для сходимости аналогов ФЧ к их оценкам алгоритму (6) – (8) требуется в среднем от 10 до 14 итераций.

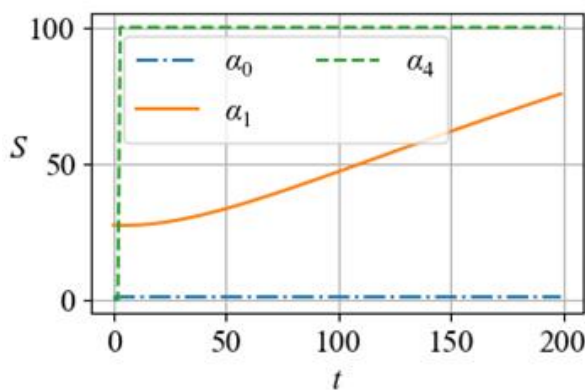


Рис.2А

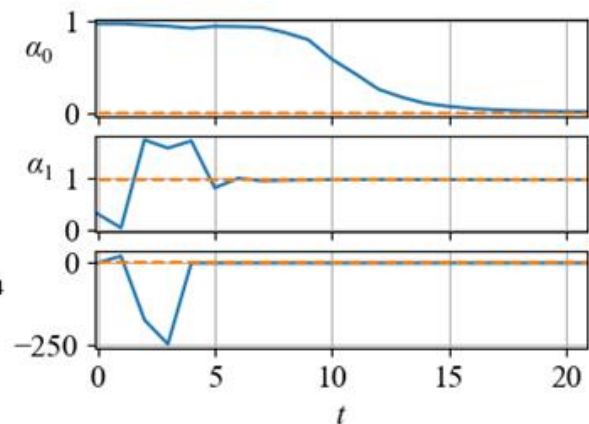


Рис.2Б

Рис.2. Оценка ФЧ с помощью адаптивного алгоритма (Рис.2А) в процессе идентификации (Рис.2Б) при использовании адаптивного МНК

Влияние аддитивных центрированных нормально распределенных помех ($e(t) \sim N(0, \sigma^2)$) на измерения выхода объекта не существенно влияют на качество оценки ФЧ алгоритмом (6) – (8). Уровень помех определяется как соотношение $NSR = A_n/A_s$, где $A_n = 3\sigma$ – амплитуда шума, A_s – амплитуда сигнала. Для модели (11) при $t \in [0, 200]$ $A_s = 48.44$. Значение критерия MSE при $NSR = 0.1$, что соответствует 10%, для всех ФЧ остается в районе $4.63 \cdot 10^{-8}$. Несмотря на это, качество идентификации ожидаемо ухудшается.

В работе рассматривается задача приближенного вычисления функций чувствительности для целей идентификации. Приводятся алгоритмы ретроспективной и адаптивной оценки функций чувствительности. Используемый подход основан на комбинированном использовании метода конечных разностей из области локального анализа чувствительности и исследовании широкой параметрической области, что характерно для глобального анализа. Структура предлагаемого алгоритма аналогична структуре алгоритмов МНК, что позволяет вычислять ФЧ в процессе накопления информации об объекте. Помимо этого преимуществом является небольшие вычислительные затраты, необходимые для вычисления выхода модели.

Разработанные методы применяются в задаче идентификации параметров модели процесса простой дистилляции, который представляет собой процесс нагрева жидкости с чистым запаздыванием и сильной инерционностью. Имитационные испытания на наглядном примере из литературы хорошо продемонстрировали эффективность предлагаемых методов.

Список литературы

1. Liu S. Global Sensitivity Analysis: The Primer by Andrea Saltelli, Marco Ratto, Terry Andres, Francesca Campolongo, Jessica Cariboni, Debora Gatelli, Michaela Saisana, Stefano Tarantola // *International Statistical Review*. 2008. Vol. 76, № 3. 452–452 p.
2. Ferretti F., Saltelli A., Tarantola S. Trends in sensitivity analysis practice in the last decade // *Science of the Total Environment*. British Geological Survey, NERC, 2016. Vol. 568. P. 666–670.
3. Douglas-Smith D. et al. Certain trends in uncertainty and sensitivity analysis: An overview of software tools and techniques // *Environmental Modelling & Software*. Elsevier Ltd, 2020. Vol. 124, № June 2019. P. 104588.
4. Turányi T. Sensitivity analysis of complex kinetic systems. Tools and applications // *Journal of Mathematical Chemistry*. 1990. Vol. 5, № 3. P. 203–248.
5. De Pauw D.J.W., Vanrolleghem P.A. Practical aspects of sensitivity function approximation for dynamic models // *Mathematical and Computer Modelling of Dynamical Systems*. 2006. Vol. 12, № 5. P. 395–414.
6. Sobol I.M. Global sensitivity indices for the investigation of nonlinear mathematical models // *Mathematical Models and Computer Simulations*. 2005. Vol. 17, № 9. P. 43–52.
7. Рубан А.И. Адаптивные системы управления с идентификацией. Красноярск: СФУ, 2015. 140 p.
8. Рубан А.И. Метод глобальной оптимизации, основанный на селективном усреднении координат, при наличии ограничений // *Вестник томского гос. ун-та. Управление, вычислительная техника и информатика*. 2013. Vol. 1, № 22. P. 114–123.
9. Chouakri N. et al. Parameter estimation of a monod-type model. Part I: Theoretical identifiability and sensitivity analysis // *Biotechnology Techniques*. 1994. Vol. 8, № 10. P. 683–688.

ПОДБОР АРХИТЕКТУРЫ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ ДЛЯ ПРЕДСКАЗАНИЯ ПОСЛЕДСТВИЙ ИНФАРКТА МИОКАРДА

С.Е. Головенкин, С.Ю. Никулина, Ю.В. Орлова, Е.Ю. Пелипецкая

Красноярский государственный медицинский университет. имени профессора

В.Ф.Войно-Ясенецкого

gse2008@mail.ru

М. Г. Доппер

Сибирский государственный университет науки и технологий им. М.Ф. Решетнева,

dorrer_mg@sibsau.ru

Введение

Инфаркт миокарда является одним из опасных заболеваний. Широкое распространение этого заболевания за последние полвека сделало его одной из самых острых проблем современной медицины. Это подтверждается статистикой, приведенной например в [1]. В развивающихся странах случаев смерти от ССЗ происходит в два раза больше, чем в развитых странах [2]. Сердечно-сосудистые заболевания являются основной причиной смерти в Великобритании, в 2003 году на них приходилось чуть менее 233 000 человек. Более 1 из 3 человек (38%) умирает от ССЗ [3]. Основными формами ССЗ являются ишемическая болезнь сердца (ИБС) и инсульт. Около половины всех случаев смерти от ССЗ - от ИБС, а около четверти – от инсульта [3].

Распространенным подходом является прогнозирование течения ИМ на основе клинико-лабораторных и клинико-инструментальных данных. Зачастую для прогноза используются клинические данные, обработанные с помощью различных математических методов. В частности, были предложены методы прогнозирования с вычислением прогностических индексов (А. Peel [4], R.Norris [5]), применялись дискриминантный анализ (J.Willems [6]), делались попытки применения алгоритмов распознавания образов. Однако, классические статистические методы характеризуются сложностью и громоздкостью некоторых прогностических правил, использованием для прогнозирования результатов наблюдения за больным на протяжении всего периода болезни, вплоть до момента развития осложнения, невозможностью дообучения и совершенствования прогностических экспертных систем на новых данных, а также использованием для прогнозирования сложных дорогостоящих диагностических исследований, проведение которых возможно далеко не во всех клиниках.

Все эти проблемы обусловили интерес к применению для предсказания последствий инфаркта миокарда искусственных нейронных сетей. Такие попытки предпринимаются начиная с начала 90-х годов (А. Zelicoff [7], D. Rossiev et al. [8], A. Gorban et al. [9], A. Borisov et al. [10]) и продолжают в настоящее время (Eggers K. et al. [11], Nag P. et al. [12]). В настоящее время работы расширены на область интерпретации кардиограмм (см., например, N. Strodthoff [13]).

Течение заболевания у больных ИМ отличается. ИМ может протекать без осложнений или с осложнениями, которые не ухудшают долгосрочный прогноз. В то же время около половины пациентов в остром и подостром периодах имеют осложнения, приводящие к ухудшению течения заболевания и даже смерти. Даже опытный специалист не всегда может предвидеть развитие этих осложнений. В связи с этим прогнозирование осложнений инфаркта миокарда с целью своевременного проведения необходимых профилактических мероприятий представляется важной задачей.

Применяемые методы исследования и экспериментальные данные

Данное исследование строится на использовании базы данных осложнений инфаркта миокарда [14]. Эта база содержит данные историй болезней больных,

перенесших инфаркт миокарда с различными осложнениями. База собрана в 1992-1995 гг. в Красноярске и снабжена соответствующим описанием.

Прогнозирование выполнялось при помощи системы Matlab 2016, модуль Neural Network Toolbox. В ходе исследования выполнялся подбор архитектуры нейронной сети, наилучшим образом предсказывающей исходы, описанные в базе данных.

Ставилась задача предсказания следующих исходов заболевания, закодированных в полях базы данных [14]:

1. A_V_BLOK – (Third-Degree Atrioventricular Block ECG);
2. DRESSLER – Синдром Дресслера (Dressler syndrome);
3. FIBR_JELUD – Фибрилляция желудочков (Ventricular fibrillation);
4. FIBR_PREDS – Мерцательная аритмия (Atrial fibrillation);
5. JELUD_TAH – Желудочковая тахикардия (Ventricular tachycardia);
6. LET_IS – Смертельный исход (Lethal outcome);
7. OTEK_LANC – Отек легких (Pulmonary edema);
8. P_IM_STEN – Постинфарктная стенокардия (Post-infarction angina);
9. PREDS_TAH – Суправентрикулярная тахикардия (Supraventricular tachycardia);
10. RAZRIV – Разрыв миокарда (Myocardial rupture);
11. REC_IM – Рецидив инфаркта миокарда (Relapse of the myocardial infarction);
12. ZSN – Хроническая сердечная недостаточность (Chronic heart failure).

Во всех экспериментах для прогнозирования использовались искусственные нейронные сети прямого распространения (функция *feedforwardnet*). Нейросеть, создаваемая при помощи этой функции, содержит набор из нескольких слоев. Первый (входной) слой связывает сеть с вектором входных данных. Далее располагаются несколько скрытых слоев нейронов, причем каждый последующий слой соединен с предыдущим. Последний (выходной) слой производит расчет одного либо нескольких выходных сигналов сети. Выполнялось обучение нейронной сети с учителем с использованием функции обучения *trainngdx*, выполняющей обучение по методу обратного распространения ошибки с оптимизацией функции ошибки методом градиентного спуска.

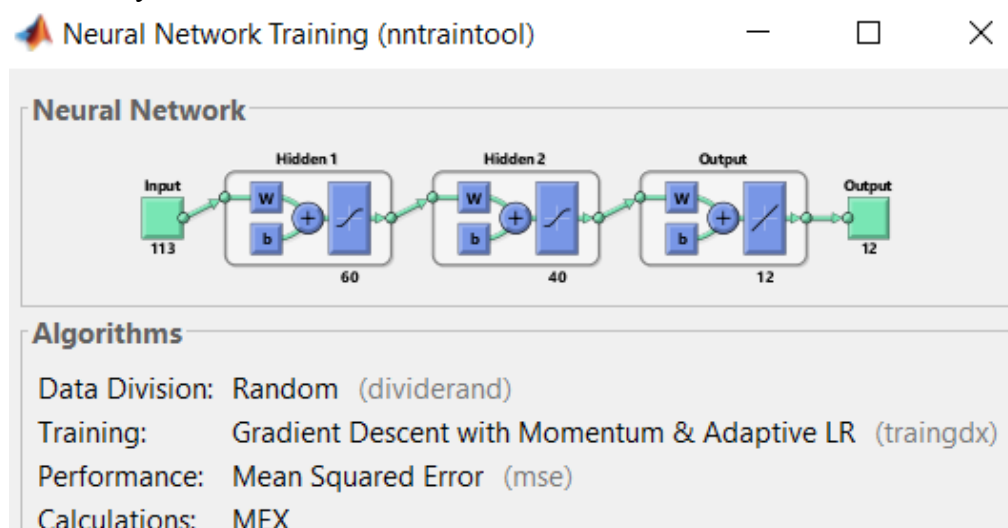


Рис. 1. Пример архитектуры нейронной сети Matlab, предсказывающей вектор из 12 диагнозов

Архитектура сети варьировалась для поиска наилучшей с точки зрения задачи предсказания исхода заболевания. Изменению подвергались:

- Количество скрытых слоев нейронов – от 1 до 3;
- Количество нейронов в каждом слое – от 10 до 100;

- Функция активации нейронов скрытых слоев и выходного слоя, при этом применялись функции:
 - *tansig* – гиперболический тангенс;
 - *logsig* – сигмоида;
 - *purelin* – линейная функция.

Варьированию также подвергался набор входных сигналов. С целью повышения качества прогноза из набора входных сигналов исключались наиболее «разреженные» столбцы, для которых количество отличных от «0» значений было меньше заданного порога.

Постановка диагноза нейросетью определялась следующим образом. Поскольку наличие соответствующего диагноза в истории болезни кодировалось в баз значением «1», а отсутствие диагноза – значением «0», принималось, что если выходной сигнал нейрона превышал значение «0.5», то диагноз считался поставленным, а если не превышал – то принималось, что соответствующий диагноз отсутствует.

Кроме того, был применен традиционный для медицинских приложений искусственных нейронных сетей метод предсказания диагноза несколькими сетями, сгенерированными с различной стартовой разметкой весов синапсов. Такой подход называется «консилиум». Взвешивание мнений нескольких сетей может выполняться путем осреднения соответствующих выходных сигналов. В данной работе был использован метод голосования.

С целью повышения эффективности использования данных используемой базы данных обучение производилось методом «скользящего окна». Данный метод заключается в исключении из обучающей выборки лишь одного примера исходной базы данных и тестировании обученной нейронной сети на исключенном примере. Таким образом, для одного сеанса необходимо обучение числа нейронных сетей, равного числу примеров обучающей выборки.

Результаты

На первом этапе производилось исследование качества предсказания нейросетями различной архитектуры. Был произведен расчет предсказания всех 12 диагнозов в базе [14]. Наиболее проблемным для предсказания стал диагноз «Хроническая сердечная недостаточность» («Chronic heart failure»). Данную тенденцию можно видеть в Таблице 1. Для данного диагноза была предпринята попытка уменьшить ошибку предсказания путем подбора архитектуры нейронной сети.

В рамках решения задачи данной статьи был выполнен поиск возможности повысить вероятность правильного предсказания за счет следующих факторов:

- Двухслойная нейросеть (по 50 нейронов в каждом из двух скрытых слоев) с исключением столбцов входных данных, включающих не более 40 случаев на выборку. Точность предсказания 66,88%.
- Двухслойная нейросеть (по 50 нейронов в каждом из двух скрытых слоев), действующая методом голосования консилиума 5 нейросетей на исходной выборке. Точность предсказания 65,97%
- Двухслойная нейросеть (по 50 нейронов в каждом из двух скрытых слоев), действующая методом голосования консилиума 5 нейросетей с исключением

столбцов входных данных, включающих не более 40 случаев на выборку. Точность предсказания 70,52%

- Нейронная сеть с одним скрытым слоем из 100 нейронов, действующая методом голосования консилиума 5 нейросетей на исходной выборке. Точность предсказания 66,72%
- Нейронная сеть с одним скрытым слоем из 100 нейронов, действующая методом голосования консилиума 5 нейросетей с исключением столбцов входных данных, включающих не более 40 случаев на выборку. Точность предсказания 68,99%
- Нейронная сеть с одним скрытым слоем из 20 нейронов, действующая методом голосования консилиума 5 нейросетей на исходной выборке. Точность предсказания 66,95%.
- Нейронная сеть с одним скрытым слоем из 20 нейронов, действующая методом голосования с исключением столбцов входных данных, включающих не более 40 случаев на выборку. Точность предсказания 68,68%.

Предварительный анализ показал, что двуслойные сети дают чуть лучший результат, чем однослойные, а применение трехслойной сети приращению качества прогноза не дало. Поэтому дальнейший анализ сосредоточился на подборе конфигурации двухслойной нейронной сети. Был произведен расчет 25 наборов нейронных сетей, отличающихся количеством нейронов в первом (L1) и втором (L2) скрытых слоях нейронной сети. Результаты представлены в Таблице 1. Видно, что ни одна из архитектур не дала значительного улучшения результатов прогноза.

Таблица 1

Результат подбора оптимальной архитектуры нейронной сети для предсказания хронической сердечной недостаточности (доля верных ответов нейронной сети).

		L1				
		20	40	60	80	100
L2	20	60,51%	61,60%	61,68%	59,65%	60,27%
	40	58,59%	62,43%	60,67%	59,88%	59,84%
	60	59,92%	62,62%	61,92%	61,49%	61,21%
	80	60,98%	61,84%	62,31%	61,41%	61,02%
	100	58,83%	61,53%	61,96%	61,49%	62,58%

Наилучший результат для предсказания хронической сердечной недостаточности был получен для нейросети, содержащей 2 скрытых слоя 60 и 40 нейронов. Именно такая конфигурация была принята как базовая для дальнейшей работы.

Для такой конфигурации нейронной сети был произведен расчет прогноза всех одиннадцати диагнозов, содержащихся в базе данных [14]. При этом дополнительно варьировались следующие показатели:

- Использование консилиума сетей – проверялись варианты прогноза одиночной нейросетью и консилиумом из 5 сетей, принимающих решение методом голосования.
- Исключение столбцов входных данных, включающих не более заданного числа случаев на выборку (как правило – не более 40, в отдельных случаях – не более 20).

Был проведен расчет, позволяющий оценить влияние размера консилиума нейросетей на качество прогноза диагнозов. Результат такого расчета для консилиума из 1, 5, 10 и 15 нейросетей, работающих с полной базой данных [14] и с этой же базой, очищенной от столбцов, содержащих менее 40 отличных от 0 значений приведен в таблице 2.

Таблица 2

Успешность прогноза вектора диагнозов выходов в зависимости от числа голосующих нейронных сетей (в процентах правильных ответов нейросетей, обученных по методу скользящего окна).

Выходное поле	Для полного набора входов				Исключены разреженные входы (не более 40 ненулевых сигналов)			
	Количество голосующих нейросетей				Количество голосующих нейросетей			
	1	5	10	15	1	5	10	15
FIBR_PREDS	86,70%	89,94%	89,94%	89,94%	86,87%	89,88%	89,94%	89,99%
PREDS_TAH	95,11%	98,71%	98,82%	98,82%	95,76%	98,76%	98,76%	98,82%
JELUD_TAH	94,64%	97,47%	97,53%	97,47%	94,47%	97,47%	97,53%	97,53%
FIBR_JELUD	91,94%	95,59%	95,82%	95,76%	92,11%	95,82%	95,76%	95,82%
A_V_BLOK	93,35%	96,59%	96,65%	96,65%	93,05%	96,59%	96,65%	96,65%
OTEK_LANC	88,23%	90,58%	90,64%	90,64%	87,58%	90,52%	90,64%	90,58%
RAZRIV	93,35%	96,76%	96,82%	96,82%	92,94%	96,76%	96,76%	96,76%
DRESSLER	91,58%	95,47%	95,59%	95,53%	92,05%	95,59%	95,59%	95,59%
ZSN	74,10%	76,63%	76,87%	76,87%	73,40%	76,63%	76,81%	76,75%
REC_IM	86,82%	90,70%	90,70%	90,64%	86,40%	90,52%	90,70%	90,64%
P_IM_STEN	87,70%	91,17%	91,29%	91,29%	87,52%	91,17%	91,29%	91,23%
LET_IS	57,39%	61,68%	67,57%	64,45%	57,33%	62,92%	69,33%	64,69%

Обсуждение и выводы

Полученные результаты прогнозирования позволяют утверждать, что искусственная нейронная сеть, обученная на предыдущем клиническом опыте, обладает достаточно высокой прогностической способностью по предсказанию исходов инфаркта миокарда. В таблице 2 можно видеть, что 9 из 12 диагнозов корректно предсказываются консилиумом нейросетей более, чем в 90% случаев, еще 1 – более, чем в 80% случаев и лишь для одного диагноза не удалось получить корректность предсказания выше 68%. Таким образом, даже база скромного объема (1700 записей) позволяет обучать нейронные сети, позволяющие достаточно точно прогнозировать исходы заболевания.

Проверка гипотезы об эффективности исключения разреженных столбцов данных для повышения точности прогноза исхода заболевания в целом подтвердилась. Таблица 2 показывает, что исключение разреженных столбцов (содержащих не более 40 ненулевых значений) позволило поднять вероятность корректного прогноза на величину 1,5 – 2%.

Также эффективным оказалось применение консилиума нейросетей. Таблица 2 показывает, что переход на постановку диагнозом из 5 нейросетей вместо одной нейросети такой же архитектуры позволяют поднять вероятность корректного прогноза на величину от 2,5 до 4%.

Вместе с тем по Таблица 2 можно видеть, что линейного нарастания качества прогнозов с ростом размера консилиума не происходит. При переходе размера консилиума с 5 на 10 происходит насыщение роста точности прогноза, а переход на размер консилиума в 15 нейронных сетей практически не дает роста точности. При этом трудоемкость обучения возрастает линейно с ростом числа сетей. Для работы с базой данных [14] достаточным размером консилиумом является 5 нейронных сетей.

Исследование архитектуры нейронной сети показывает, что достаточный размер – 2 скрытых слоя по 40-60 сигмоидных нейронов.

В целом можно видеть, что все примененные подходы к повышению точности прогнозов дали прирост вероятности правильной постановки диагноза в общей совокупности от 7 до 9%. Наилучшие конфигурации консилиумов нейронных сетей обеспечили для 10 диагнозов из 12 вероятность корректной постановки диагноза до 90% и более, для двух оставшихся – 65% и 76%.

Таким образом, примененный подход к прогнозированию последствий инфаркта миокарда показал в большинстве случаев высокую точность. Следовательно, поставленная задача прогнозирования осложнений инфаркта миокарда искусственными нейронными сетями решена в данной работе успешно. Показано, что точность предсказания повышается за счет очистки исходной базы данных от сильно разреженных столбцов, а также за счет применения консилиума нейронных сетей, действующих по методу голосования.

Перспективы дальнейшей работы заключаются в следующем:

- В применении для обучения нейросети базы данных более современных, чем использованная в исследовании [14], включающей большее число входных параметров, а также большее число случаев.
- В проверке предположения, что обучающая выборка большего размера позволит обучить и более сложную по архитектуре нейронную сеть, которая, соответственно, будет обладать лучшей прогностической способностью.
- Переход к разработке нейросетевой экспертной системы для предсказания последствий инфаркта миокарда на специализированных платформах для разработки моделей нейронных сетей, например, TensorFlow/Keras

Список литературы

1. Health Knowledge. [Электронный ресурс], <https://www.healthknowledge.org.uk/public-health-textbook/disease-causation-diagnostic/2b-epidemiology-diseases-phs/chronic-diseases/coronary-heart-disease>
2. Reducing risks, Promoting Healthy Life. /. — : World Health Organization. The World Health Report 2002. Geneva, 2002.
3. 2005 Coronary Heart Disease Statistics - 2005 edition, British Heart Foundation Health Promotion Research Group, 2006. Available online at: <http://www.heartst> / Petersen, S., Peto, V., Scarborough, P., Rayner, M. — Oxford Department of Public Health, University of Oxford, 2006.

4. Peel, A. F., Semple, T., Wang, I., Lancaster, W. M., Dall, J. L.G., A coronary prognostic index for grading the severity of infarction, *Heart*, Том 24, 1962. — с. 745-760.
5. Norris, R. M., Brandt, P. W. T., Caughey, D. E., Lee, A. J., Scott, P. J., A new coronary prognostic index, *The Lancet*, February 1969.
6. Willems, J. L., Pardaens, J., De Geest, H., Early risk stratification using clinical findings in patients with acute myocardial infarction, *European Heart Journal*, 1984. — с. 130–139.
7. Zelicoff, A. P., Myocardial infarction prediction by artificial neural networks, *Annals of internal medicine*, Выпуск 116(8), 1992.
8. Rossiev, D. A., Golovenkin, S. E., Shulman, V. A., Matyushin, G. V. Forecasting of myocardial infarction complications with the help of neural networks // *World Congress on Neural Networks*. — Washington, DC, 1995.
9. Gorban, A. N., Rossiev, D. A., Butakova, E. V., Gilev, S. E., Golovenkin, S. E., Dogadin, S. A., Dorrer, M. G. et al. Medical, psychological and physiological applications of MultiNeuron neural simulator // *Second International Symposium on Neuroinformatics and Neurocomputers*. — Rostov on Don, Russia, 1995.
10. Borisov, A. G., Gilev, S. E., Golovenkin, S. E., Gorban, A. N., Dogadin, S. A., Kochenov, D. A., Maslennikova, E. V. et al., "Multineuron" neural simulator and its medical applications, *Modelling, Measurement & Control*, Том 55, Выпуск 1, 1996. — с. 1-5.
11. Eggers, Kai M., Ellenius, Johan, Dellborg, Mikael, Groth, Lars Groth, Oldgren, Jonas, Swahn, Eva, Lindahl, Bertil, Artificial neural network algorithms for early diagnosis of acute myocardial infarction and prediction of infarct size in chest pain patients, *International journal of cardiology*, Том 114(3), 2007.
12. Nag, Procheta, Mondal, Saikat, More, Arun, A Holistic User Centric Acute Myocardial Infarction Prediction System With Model Evaluation Using Data Mining Techniques, *Advances in Science, Technology and Engineering Systems Journal (ASTESJ)*, Том 3, Выпуск 6, 2018. — с. 56-66.
13. Strodthoff, Nils, Strodthoff, Claas, Detecting and interpreting myocardial infarction using fully convolutional neural networks, *Physiological Measurement*, Том 40, 2019.
14. Golovenkin, S. E., Gorban, Alexander, Mirkes, Evgeny, Shulman, V. A., Rossiev, D. A., Shesternya, P. A., Nikulina, S. Yu. et al. (2020, Mar.) University of Leicester. [Электронный ресурс], https://leicester.figshare.com/articles/Myocardial_infarction_complications_Database/12045261

ОЦЕНКА ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ПРЕДСКАЗАНИЯ ОСЛОЖНЕНИЙ ИНФАРКТА МИОКАРДА

С.Е. Головенкин, С.Ю. Никулина, Ю.В. Орлова, Е.Ю. Пелипецкая
Красноярский государственный медицинский университет. имени профессора

В.Ф.Войно-Ясенецкого

gse2008@mail.ru

М. Г. Доппер

Сибирский государственный университет науки и технологий им. М.Ф. Решетнева,
dorrer_mg@sibsau.ru

Введение

Острота проблемы сердечно-сосудистых заболеваний подтверждается многочисленными данными статистики [1]. В развитых странах сердечные заболевания и инсульт являются первой и второй по значимости причиной смерти среди взрослых мужчин и женщин, а в развивающихся странах случаев смерти от ССЗ происходит в два раза больше, чем в развитых странах [2]. В целом, в развивающихся странах ССЗ занимают третье место по доле болезней. В Великобритании по данным 2003 года на сердечно-сосудистые заболевания пришлось более трети зафиксированных смертей (38%) [3].

Настолько высокая острота проблемы заставляет искать новые подходы к профилактике, лечению и реабилитации больных с данным заболеванием. Для решения этих задач необходимо, в частности, совершенствование методов прогнозирования течения и исходов острого ИМ. Многие авторы прогнозировали течение ИМ на основе клинико-лабораторных и клинико-инструментальных данных. Зачастую для прогноза используются клинические данные, обработанные с помощью различных математических методов. В частности, были предложены методы прогнозирования с вычислением прогностических индексов (А. Peel [4], R.Norris [5]), применялись дискриминантный анализ (J.Willems [6]), делались попытки применения алгоритмов распознавания образов. Вместе с тем, нейронные сети имеют значительное преимущество перед классическим линейными или статистическими методами (например, реализованными в MS Excel [7], Statistica [8]) за счет эффективно реализованных алгоритмов самообучения [9], [10].

Однако такой подход несет и существенный минус – требование к большой выборке клинических случаев с известным исходом. Формирование архитектуры ИНС включает в себя действия как по изменению топологии ИНС, так и по подбору количества обучаемых параметров. В процессе развития технологии ИНС возник подход «deep learning» [11]. ИНС позволяют обнаруживать взаимосвязи между изучаемыми признаками, которые не всегда очевидны при изучении традиционными методами статистического анализа.

Попытки применения нейронных сетей для предсказания хода течения и последствий сердечно-сосудистых заболеваний предпринимаются начиная с начала 90-х годов (А. Zelicoff [12], D. Rossiev et al. [13], А. Gorban et al. [14]). Работы в этом направлении продолжают в настоящее время (Eggers K. et al. [15], Nag P. et al. [16]). В работе [17] авторами была использована рекуррентная ИНС для раннего выявления начала сердечной недостаточности. В настоящее время работы расширены на область интерпретации кардиограмм (см., например, N. Strodthoff [18], Hannun et al. [19], Mena et al. [20]). В работе [21] рассматривалось использование ИНС для обработки ЭКГ с точки зрения применения вейвлет преобразований. В работе [22] Sannino и De Pietro

описывают применение ИНС глубокого обучения для автоматической классификации сигналов ЭКГ. Yildirim et al. в работе [23] использовали сверточную сеть 1D-CNN для определения сердечной аритмии на данных ЭКГ большой длины.

Широкое применение получили ИНС при разработке экспертных систем, предназначенных для прогнозирования хода течения сердечно-сосудистых заболеваний.

В работе [24] описывается система, предсказывающая сроки пребывания пациентов в стационаре с одним из трех основных диагнозов: коронарный атеросклероз (КАС), сердечная недостаточность (СН) и острый инфаркт миокарда (ОИМ). Статья [25] описывает опыт прогнозирования занятости больничных коек при помощи ИНС. В работе [26] описана нейросетевая система, использовавшаяся для мониторинга за состоянием пациента и оповещения об ухудшении его состояния. Meyer et al. предложили [27] использовать рекуррентную ИНС для прогнозирования серьезных осложнений во время интенсивной терапии в режиме реального времени после кардиохирургического вмешательства на открытом сердце. Статья [28] описывает применение ИНС для прогнозирования риска заболеваний на основании демографической, физиологической и биохимической информации. Weng et al. в работе [29] при помощи ИНС решали задачу оценки рисков формирования сердечно-сосудистого заболевания.

Течение заболевания у больных ИМ отличается. ИМ может протекать без осложнений или с осложнениями, которые не ухудшают долгосрочный прогноз. В то же время около половины пациентов в остром и подостром периодах имеют осложнения, приводящие к ухудшению течения заболевания и даже смерти. Даже опытный специалист не всегда может предвидеть развитие этих осложнений. В связи с этим прогнозирование осложнений инфаркта миокарда с целью своевременного проведения необходимых профилактических мероприятий представляется важной задачей. Вместе с тем, высокие требования к объему выборки для обучения ИНС требуют применения специальных методов для повышения предсказательной способности разрабатываемых нейросетевых экспертных систем кардиологического назначения. Решению этой задачи посвящена данная статья.

Применяемые методы исследования и экспериментальные данные

В качестве экспериментальных данных, использующихся для обучения искусственной нейронной сети, в данной работе взята база данных осложнений инфаркта миокарда [30]. Эта база содержит данные историй болезней 1700 больных, перенесших инфаркт миокарда с различными осложнениями. База собрана в 1992-1995 гг. в Красноярске и снабжена соответствующим описанием.

Прогнозирование выполнялось при помощи системы Matlab 2016, модуль Neural Network Toolbox. В ходе исследования выполнялся подбор архитектуры нейронной сети, наилучшим образом предсказывающей исходы, описанные в базе данных.

Ставилась задача предсказания следующих исходов заболевания, закодированных в полях базы данных [30]:

1. A_V_BLOK – (Third-Degree Atrioventricular Block ECG);
2. DRESSLER – Синдром Дресслера (Dressler syndrome);
3. FIBR_JELUD – Фибрилляция желудочков (Ventricular fibrillation);
4. FIBR_PREDS – Мерцательная аритмия (Atrial fibrillation);
5. JELUD_TAH – Желудочковая тахикардия (Ventricular tachycardia);
6. LET_IS – Смертельный исход (Lethal outcome);
7. OTEK_LANC – Отек легких (Pulmonary edema);
8. P_IM_STEN – Постинфарктная стенокардия (Post-infarction angina);

9. PREDS_TAH – Суправентрикулярная тахикардия (Supraventricular tachycardia);
10. RAZRIV – Разрыв миокарда (Myocardial rupture);
11. REC_IM – Рецидив инфаркта миокарда (Relapse of the myocardial infarction);
12. ZSN – Хроническая сердечная недостаточность (Chronic heart failure).

Во всех экспериментах для прогнозирования использовались искусственные нейронные сети прямого распространения (функция *feedforwardnet*). Нейросеть, создаваемая при помощи этой функции, содержит набор из нескольких слоев. Первый (входной) слой связывает сеть с вектором входных данных. Далее располагаются два скрытых слоя нейронов (60 нейронов и 40 нейронов соответственно), причем каждый последующий слой соединен с предыдущим. Последний (выходной) слой производит расчет выходного сигнала сети, отвечающего за прогноз наступления одного из диагнозов. Постановка диагноза нейросетью определялась следующим образом. Поскольку наличие соответствующего диагноза в истории болезни кодировалось в баз значением «1», а отсутствие диагноза – значением «0», принималось, что если выходной сигнал нейрона превышал значение «0.5», то диагноз считался поставленным, а если не превышал – то принималось, что соответствующий диагноз отсутствует.

Выполнялось обучение нейронной сети с учителем с использованием функции обучения *traingd*, выполняющей обучение по методу обратного распространения ошибки с оптимизацией функции ошибки методом градиентного спуска. Кроме того, был применен традиционный для медицинских приложений нейронных сетей метод предсказания диагноза несколькими сетями, сгенерированными с различной стартовой разметкой весов синапсов. Такой подход называется «консилиум». Взвешивание мнений нескольких сетей может выполняться путем осреднения соответствующих выходных сигналов. В данной работе был использован метод голосования – выставлялся тот диагноз, который поставило простое большинство из голосовавших нейросетей.

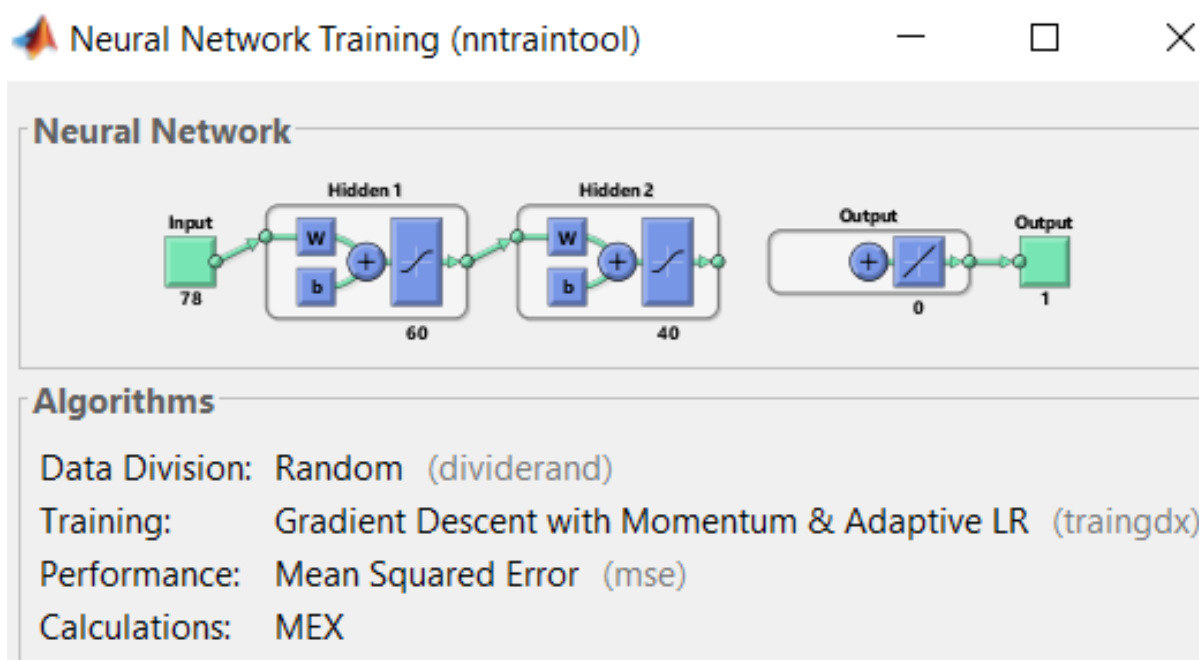


Рис. 1. Пример архитектуры нейронной сети Matlab, использовавшейся в прогнозе

С целью повышения эффективности использования данных используемой базы данных обучение производилось методом «скользящего окна». Данный метод заключается в исключении из обучающей выборки лишь одного примера исходной базы данных и тестировании обученной нейронной сети на исключенном примере. Таким образом, для одного сеанса необходимо обучение числа нейронных сетей, равного числу примеров обучающей выборки.

Качество предсказания нейросетью последствий инфаркта миокарда оценивалось стандартными показателями, предназначенными для оценки эффективности диагностических методик – Точность и Специфичность.

Точность – показатель, позволяющий сопоставить различные методы определения одного показателя при обследовании одной популяции больных. Представляет собой число истинных результатов к числу всех результатов обследования.

Чувствительный тест следует выбрать, если есть риск пропустить заболевание при неясной диагностической картине или сузить рамки диагностического поиска, исключив ряд частых причин с помощью высокочувствительных тестов. Чувствительность представляет собой число корректных положительных результатов обследования к общему числу обследованных больных.

Специфичность указывает на число корректных отрицательных результатов обследования к общему числу контрольной группы. Специфичные тесты нужны для подтверждения диагноза, предложенного на основании других данных. Высокоспецифичные тесты особенно необходимы, если ложноположительный результат может нанести пациенту вред, например, в результате ошибочно назначенного лечения.

Полученные результаты

Для такой конфигурации нейронной сети был произведен расчет прогноза всех двенадцати диагнозов, содержащихся в базе данных [30]. При этом дополнительно варьировались следующие показатели:

- Использование консилиума сетей – проверялись варианты прогноза одиночной нейросетью, а также консилиумом из 3 сетей, принимающих решение методом голосования.
- Исключение наиболее разреженных столбцов входных данных, включающих не более 40 случаев на общую выборку из 1700 случаев.

Результаты расчета для точности, чувствительности и специфичности прогноза последствий инфаркта миокарда консилиумами искусственных нейронных сетей приведены соответственно в таблицах 1 и 2.

Таблица 1

Точность предсказания последствий инфаркта миокарда ИНС.

	Contrast	All		40+	
	K=	1	3	1	3
Диагноз	FIBR_PREDS	87,93%	89,94%	87,46%	89,17%
	PREDS_TAH	97,76%	98,76%	97,41%	98,71%
	JELUD_TAH	96,29%	97,47%	96,41%	97,41%
	FIBR_JELUD	94,53%	95,64%	94,23%	95,70%
	A_V_BLOK	95,82%	96,59%	94,76%	96,47%
	OTEK_LANC	89,64%	90,11%	89,05%	90,46%
	RAZRIV	95,47%	96,70%	95,29%	96,70%
	DRESSLER	94,06%	95,53%	93,88%	95,47%
	ZSN	75,16%	75,57%	73,16%	75,46%

	REC_IM	88,82%	90,64%	88,29%	90,46%
	P_IM_STEN	89,35%	91,23%	89,58%	90,99%
	LET_IS	77,40%	81,81%	73,69%	79,87%

Таблица 2

Специфичность предсказания последствий инфаркта миокарда ИНС.

	Contrast	All		40+	
	K=	1	3	1	3
Диагноз	FIBR_PREDS	90,17%	90,18%	90,27%	90,01%
	PREDS_TAH	98,81%	98,82%	98,81%	98,82%
	JELUD_TAH	97,50%	97,53%	97,56%	97,53%
	FIBR_JELUD	95,82%	95,81%	95,86%	95,82%
	A_V_BLOK	96,67%	96,64%	96,64%	96,64%
	OTEK_LANC	90,89%	90,59%	91,13%	90,77%
	RAZRIV	96,78%	96,82%	96,77%	96,82%
	DRESSLER	95,57%	95,58%	95,51%	95,58%
	ZSN	78,36%	77,37%	78,21%	77,54%
	REC_IM	90,66%	90,79%	90,61%	90,63%
	P_IM_STEN	91,22%	91,28%	91,44%	91,26%
	LET_IS	85,40%	85,14%	85,24%	85,38%

Обсуждение и выводы

Полученные результаты прогнозирования позволяют утверждать, что искусственная нейронная сеть, обученная на предыдущем клиническом опыте, обладает достаточно высокой прогностической способностью по предсказанию исходов инфаркта миокарда. В таблице 2 можно видеть, что 9 из 12 диагнозов предсказываются нейросетью выбранной архитектуры с точностью свыше 90% и лишь один – с точностью менее 80%. При этом показатель специфичности диагноза (Таблица 2) сохраняется практически неизменным. Таким образом, даже база достаточно скромного объема (1700 записей) позволяет обучать нейронные сети, позволяющие достаточно точно прогнозировать исходы заболевания.

В отличие от ранее проведенных исследований не подтвердилась гипотеза об эффективности исключения разреженных столбцов данных для повышения точности прогноза исхода заболевания. Таблица 2 показывает, что исключение разреженных столбцов (содержащих не более 40 ненулевых значений) практически не повлияло на точность прогноза.

Подтвердилась эффективность применения консилиума нейросетей. Таблица 2 показывает, что переход на постановку диагноза консилиумом из 3 нейросетей вместо одной нейросети такой же архитектуры позволяют поднять точность прогноза на 1-2%.

В целом можно видеть, что все примененные подходы к повышению точности прогнозов дали прирост вероятности правильной постановки диагноза в общей совокупности от 2%. Наилучшие конфигурации консилиумов нейронных сетей обеспечили для 3 диагнозов из 12 точность около 95% при специфичности до 98%. Точность более 90% достигнута для 10 диагнозов из 12, наихудший результат – 80% точность при специфичности 85%.

Таким образом, исследование качества прогноза предсказания последствий инфаркта миокарда при помощи нейронных сетей показывает достаточно высокие точность и специфичность данного метода. Можно видеть, что применение консилиума нейронных сетей, действующих по методу голосования, повышает точность прогноза для всех диагнозов.

Перспективы дальнейшей работы заключаются в следующем:

- В применении для обучения нейросети базы данных более современных, чем использованная в исследовании [30], включающей большее число входных параметров, а также большее число случаев. Следует стремиться к получению более сбалансированной базы данных, включающей более равномерно примеры наличия и отсутствия диагноза.
- В проверке кроме перечисленных показателей качества прогноза еще и показателя чувствительности, оценивающего соотношение корректных и некорректных положительных тестов.
- В переходе к разработке нейросетевой экспертной системы для предсказания последствий инфаркта миокарда на специализированных платформах для разработки моделей нейронных сетей, например, TensorFlow/Keras

Список литературы

1. Coronary heart disease [Electronic resource]. URL: <https://www.healthknowledge.org.uk/public-health-textbook/disease-causation-diagnostic/2b-epidemiology-diseases-phs/chronic-diseases/coronary-heart-disease>.
2. The world health report 2002 - Reducing Risks, Promoting Healthy Life.
3. Allender S. et al. Coronary heart disease statistics. 2006.
4. Peel A.A.F. et al. A CORONARY PROGNOSTIC INDEX FOR GRADING THE SEVERITY OF INFARCTION // Heart. 1962. Vol. 24, № 6. P. 745–760.
5. Norris R.M. et al. A new coronary prognostic index // Am. Heart J. 1970. Vol. 79, № 3. P. 428–431.
6. WILLEMS J.L., PARDAENS J., DE GEEST H. Early risk stratification using clinical findings in patients with acute myocardial infarction // Eur. Heart J. 1984. Vol. 5, № 2. P. 130–139.
7. Herkenhoff L., Fogli J. Applied Statistics for Business and Management using Microsoft Excel. New York, NY: Springer New York, 2013.
8. Weiß C.H. StatSoft, Inc., Tulsa, OK.: STATISTICA, Version 8 // AStA Adv. Stat. Anal. 2007. Vol. 91, № 3. P. 339–341.
9. Grapov D. et al. Rise of Deep Learning for Genomic, Proteomic, and Metabolomic Data Integration in Precision Medicine // Omi. A J. Integr. Biol. 2018. Vol. 22, № 10. P. 630–636.
10. Han S.-H. et al. Artificial Neural Network: Understanding the Basic Concepts without Mathematics // Dement. Neurocognitive Disord. 2018. Vol. 17, № 3. P. 83.
11. Goodfellow I., Bengio Y., Courville A. Deep Learning. The MIT Press, 2016.
12. Browner W. Myocardial infarction prediction by artificial neural networks // Ann. Intern. Med. 1992. Vol. 116. P. 701; author reply 702.
13. Rossiev D.A. et al. Neural networks for forecasting of myocardial infarction complications // The Second International Symposium on Neuroinformatics and Neurocomputers. IEEE. P. 292–298.

14. Gorban A.N. et al. Medical, psychological and physiological applications of MultiNeuron neural simulator // International RNNS/IEEE Symposium on Neuroinformatics and Neurocomputers. 1995.
15. Eggers K.M. et al. Artificial neural network algorithms for early diagnosis of acute myocardial infarction and prediction of infarct size in chest pain patients // Int. J. Cardiol. 2007. Vol. 114, № 3. P. 366–374.
16. Nag P., Mondal S., More A. A Holistic User Centric Acute Myocardial Infarction Prediction System With Model Evaluation Using Data Mining Techniques // Adv. Sci. Technol. Eng. Syst. J. 2018. Vol. 3, № 6. P. 56–66.
17. Choi E. et al. Using recurrent neural network models for early detection of heart failure onset // J. Am. Med. Informatics Assoc. 2016. P. ocw112.
18. Strodthoff N., Strodthoff C. Detecting and interpreting myocardial infarction using fully convolutional neural networks // Physiol. Meas. 2019. Vol. 40, № 1. P. 015001.
19. Hannun A.Y. et al. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network // Nat. Med. 2019. Vol. 25, № 1. P. 65–69.
20. Mena L.J. et al. Mobile Personal Health Monitoring for Automated Classification of Electrocardiogram Signals in Elderly // Comput. Math. Methods Med. 2018. Vol. 2018. P. 1–9.
21. Kalyakulina A.I. et al. Finding Morphology Points of Electrocardiographic-Signal Waves Using Wavelet Analysis // Radiophys. Quantum Electron. 2019. Vol. 61, № 8–9. P. 689–703.
22. Sannino G., De Pietro G. A deep learning approach for ECG-based heartbeat classification for arrhythmia detection // Futur. Gener. Comput. Syst. 2018. Vol. 86. P. 446–455.
23. Yıldırım Ö. et al. Arrhythmia detection using deep convolutional neural network with long duration ECG signals // Comput. Biol. Med. 2018. Vol. 102. P. 411–420.
24. Tsai P.-F. (Jennifer) et al. Length of Hospital Stay Prediction at the Admission Stage for Cardiology Patients Using Artificial Neural Network // J. Healthc. Eng. 2016. Vol. 2016. P. 1–11.
25. Kutafina E. et al. Recursive neural networks in hospital bed occupancy forecasting // BMC Med. Inform. Decis. Mak. 2019. Vol. 19, № 1. P. 39.
26. Kwon J. et al. An Algorithm Based on Deep Learning for Predicting In-Hospital Cardiac Arrest // J. Am. Heart Assoc. 2018. Vol. 7, № 13.
27. Meyer A. et al. Machine learning for real-time prediction of complications in critical care: a retrospective study // Lancet Respir. Med. 2018. Vol. 6, № 12. P. 905–914.
28. Nanayakkara S. et al. Characterising risk of in-hospital mortality following cardiac arrest using machine learning: A retrospective international registry study // PLOS Med. / ed. Saria S. 2018. Vol. 15, № 11. P. e1002709.
29. Weng S.F. et al. Can machine-learning improve cardiovascular risk prediction using routine clinical data? // PLoS One / ed. Liu B. 2017. Vol. 12, № 4. P. e0174944.
30. Golovenkin S.E. et al. Myocardial infarction complications Database. University of Leicester, 2020.

РАЗРАБОТКА СИСТЕМЫ РАСПОЗНАВАНИЯ КОРОВ И ОПРЕДЕЛЕНИЯ УРОВНЯ ИХ АКТИВНОСТИ НА ОСНОВЕ ПРИМЕНЕНИЯ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

М.В. Гунер, И.А. Бусс

Алтайский государственный технический университет им. И.И. Ползунова,
каф. информационных систем в экономике, horyzont1@mail.ru

Современные технологии воспроизводства крупного рогатого скота основаны на программах искусственного осеменения и требуют тщательного контроля физиологического состояния животных: температуры тела, уровня активности и т.д. Наблюдательный подход за животными, в частности за коровами, со стороны человека имеет важный недостаток в виде высокой вероятности пропуска «скрытой охоты».

Как правило, коровы приходят в охоту через 50-60 дней после отела, хотя летом этот показатель находится в пределах 30-45 дней, а зимой увеличивается до 65-70 дней. Несвоевременность обнаружения коровы в состоянии «охоты» влечет за собой существенное снижение надоев молока, а впоследствии риск полной непригодности коровы для дальнейшего воспроизводства [1, 2].

Цель работы – разработать на основе применения сверточных нейронных сетей систему распознавания и идентификации коров по изображениям с веб-камеры, а также определения уровня активности животных.

В качестве исходных данных в работе был взят видеофайл, полученный с веб-камеры одного из сельскохозяйственных предприятий Алтайского края. Видеоряд был разбит на кадры, изображения коров с этих кадров легли в основу обучающей выборки. Алгоритм получения коллекций изображений с разными коровами приведен ниже.

1. Загрузка видеофайла с веб-камеры и извлечение кадров
2. Текущий кадр = 1
3. Обнаружить коров (корову) на кадре
4. Если на текущем кадре не обнаружена ни одна корова, то перейти к шагу 10
5. Номер коровы = 1
6. Получить координаты прямоугольной области с изображением коровы
7. Сохранить фрагмент кадра с изображением коровы в отдельный файл .jpg
8. Номер коровы = Номер коровы + 1
9. Если Номер коровы <= Количеству коров на кадре, то перейти к шагу 6
10. Текущий кадр = Текущий кадр + 1
11. Если Текущий кадр <= Последний кадр, то перейти к шагу 3.
12. Разделение каталога файлов .jpg с изображениями коров на подкаталоги (каждый подкаталог содержал изображения лишь одной коровы, имя подкаталога – кличка коровы)

В настоящей работе были использованы:

- язык программирования Python (Python 3.7 x64);
- библиотека компьютерного зрения opencv-python 4.3.0.36;
- библиотеки машинного обучения tensorflow 1.14, keras 2.1.2, imageai 2.1.5;
- модель обнаружения объектов RetinaNet, resnet50_coco_best_v2.0.1.h5;
- библиотека numpy 1.16;
- среда разработки PyCharm.

Модель обнаружения объектов (RetinaNet, resnet50_coco_best_v2.0.1.h5), основанная на сверточных нейронных сетях и поддерживаемая ImageAI, может обнаруживать 80 различных типов объектов: person, bicycle, car, motorcycle, airplane, bus, train, truck, boat, traffic light, fire hydrant, stop_sign, parking meter, bench, bird, cat, dog, horse, sheep, **cow**, elephant, bear, zebra, giraffe, backpack, umbrella, handbag, tie, suitcase, frisbee, skis, snowboard, sports ball, kite, baseball bat, baseball glove, skateboard, surfboard, tennis racket, bottle, wine glass, cup, fork, knife, spoon, bowl, banana, apple, sandwich, orange, broccoli, carrot, hot dog, pizza, donut, cake, chair, couch, potted plant, bed, dining table, toilet, tv, laptop, mouse, remote, keyboard, cell phone, microwave, oven, toaster, sink, refrigerator, book, clock, vase, scissors, teddy bear, hair dryer, toothbrush.

Отметим, что ImageAI позволяет выполнять обнаружение одного, нескольких или всех типов объектов из списка выше. В настоящей задаче нас интересовало обнаружение на изображении коров (тип объекта: cow). Параметр detection_speed позволил настроить соотношение скорости и качества обнаружения животных [3, 4].

Таким образом, мы получили обучающую выборку для последующей задачи идентификации животного (распознавание конкретной коровы). Фрагмент обучающей выборки представлен на рисунке 1, общие сведения – в таблице 1.

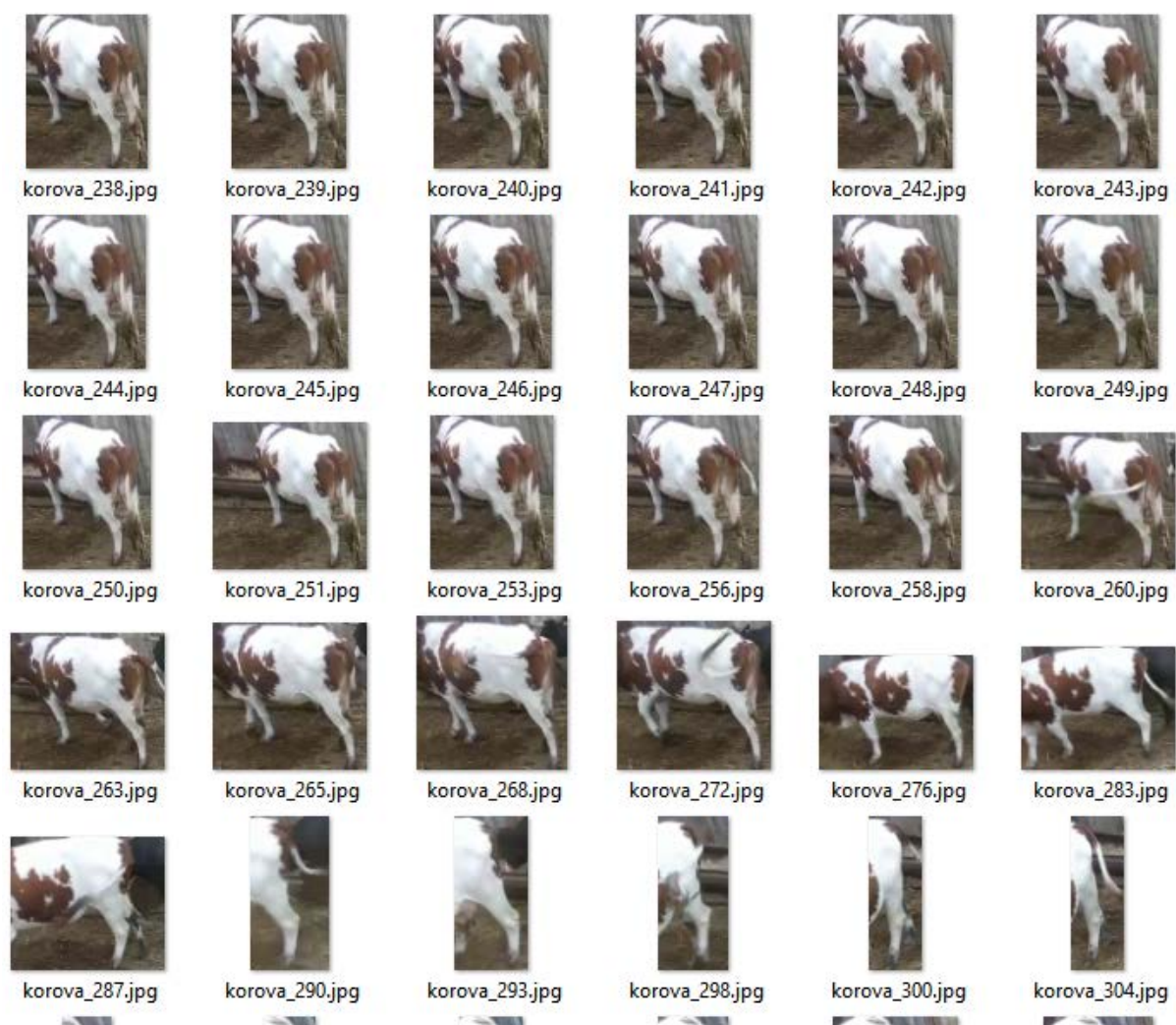


Рис. 1. Фрагмент обучающей выборки. Изображения коровы по кличке Фейя

Таблица 1

Общие сведения об обучающей выборке

№	Кличка коровы	Количество изображений	Фотоизображение коровы
1	Клюква	132	
2	Мия	229	
3	Фея	374	
4	Фрося	141	
5	Шима	436	

Для идентификации коровы (1 - Клюква, 2 – Мия, 3 – Фея, 4 - Фрося, 5 – Шима...) используем механизм свертки изображений, т.е. обработки растровых изображений путем наложения на них фильтров, отвечающих за те или иные признаки [3].

Алгоритм идентификации коровы выглядит следующим образом:

- «сворачиваем» эталонные изображения (изображения каждой коровы из обучающей выборки);

- обнаруживаем с помощью библиотек машинного зрения и машинного обучения на новом изображении, на кадре с веб-камеры (из видеофайла) новые объекты типа cow (корова), передаем фрагменты изображения с новыми объектами типа cow (корова) на следующий шаг;

- «сворачиваем» изображения с новыми (неизвестными) объектами типа cow (корова), полученные на предыдущем шаге;

- сравниваем результат свертки с результатами свертки эталонных изображений, находим минимальное расхождение, определяем номер и кличку коровы.

В настоящей работе мы использовали 4 фильтра, кодирующих признак «цветовой окрас коровы», каждый фильтр представлял собой файл .jpg разрешением 7 пикселей * 7 пикселей (рисунок 2). Перед наложением фильтров каждое изображение было обработано в части удаления фона.

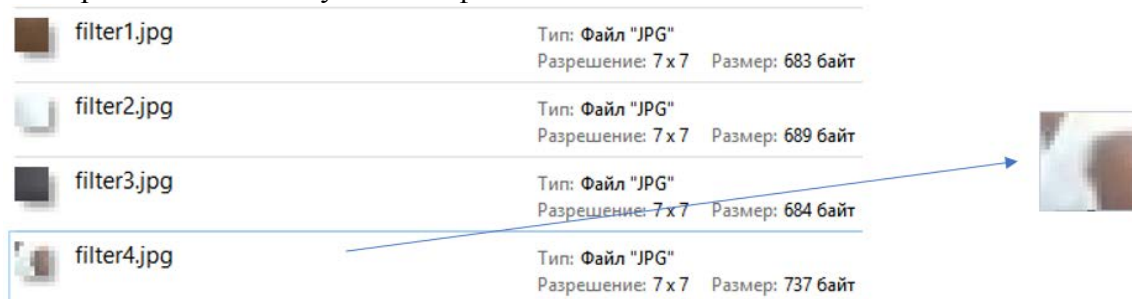


Рис. 2. Фильтры, используемые при свертке изображений коров

Работа системы распознавания и идентификации коров показана на рисунке 3. Сдвигание во времени прямоугольной рамки, очерчивающей область с изображением той или иной коровы, условно характеризовало активность животного (на рисунке, представленном ниже, временной интервал между анализом (i)-ого и (i+1)-ого изображения составлял 5 секунд).

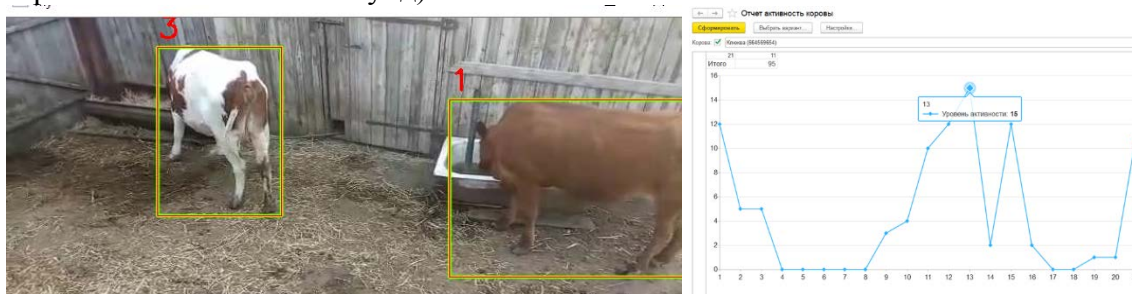


Рис. 3. Результат работы системы распознавания и идентификации коров, определения уровня их активности

Тестирование системы показало, что примененные в работе алгоритмы для распознавания и идентификации животного достаточно эффективные. Точность обнаружения и идентификации коровы составила 92 %. Однако, работу следует продолжать, так как окрас одной коровы от окраса другой коровы может отличаться весьма незначительно, и необходим более глубокий анализ «рубашек» коров.

Список литературы

1. Эффективная программа искусственного осеменения стада крупного рогатого скота [Электрон.] Режим доступа: <https://agrofid.ru/2018/07/16/artificial-insemination/>
2. Пути интенсификации воспроизводства стада в скотоводстве / А.В. Глаз и др. – Гродно: ГГАУ, 2011. - 80 с.
3. Нейронные сети и компьютерное зрение. Сверточные нейронные сети. Режим доступа: <https://stepik.org/course/50352/syllabus>
4. ImageAI. Detection Classes. Режим доступа: <https://imageai.readthedocs.io/en/latest/detection/>

КЛАСТЕРНЫЙ АНАЛИЗ ВЫСОКОРАЗМЕРНЫХ ДАННЫХ В ЗАДАЧЕ КЛАССИФИКАЦИИ ОБРАЗОВ

А.Ю. Дорогов

Санкт-Петербургский государственный электротехнический университет,
vaksa2006@yandex.ru

Введение

Классификация образов обычно трактуется как отнесение объекта к тому или иному классу из заданного множества классов, где под классом понимается некоторая категория объектов, обладающих рядом общих свойств, а под образом – представление объекта в признаковом пространстве классификатора [1]. Построить классификатор — это значит предложить модель пригодную для распознавания объектов, то есть построить алгоритм, способный для любого объекта по его образу выдать ответ о принадлежности к классу. Одним из наиболее распространенных способов построения классификаторов является машинное обучение. В этой постановке известна только конечная совокупность пар «объект, класс», называемая обучающей выборкой. На основе этих данных требуется построить классификатор, способный выдать достаточно точный ответ для любого возможного входного объекта.

Достоверность распознавания объектов зависит от пространственной конфигурации классов, которая по существу и определяет выбор подходящего алгоритма классификации. Оценкой конфигурации может служить кластерная структура класса, т.е. декомпозиция множества точек класса на подмножества. Выявление кластерной структуры является предметом кластерного анализа. Таким образом, задача построения надёжного классификатора коррелируется с задачей выявления кластерной структуры классов.

В данной работе предлагается метод эффективного выделения кластерной структуры в высокоразмерной обучающей выборке большого объема. Метод позволяет без использования метрики расстояния, локализовать центральные кластерные зоны (центры) и моды кластеров в подпространствах признакового пространства, и пригоден для анализа как числовых, так и категориальных данных.

Связанные работы

Следуя гипотезе компактности, под кластером будем понимать область признакового пространства, в которой плотность точек образов выше, чем в её ближайшей окрестности (см. рис.). В зависимости от принятой модели кластера приходят к различным методам кластерного анализа. В работе [2] представлен известный метод DBSCAN. Для этого метода модель кластера определяется через связанные цепочки локальных областей с малым радиусом ε , в которых количество точек образов превышает заданный порог. Базовая идея DBSCAN была развита в методе OPTICS [3]. В этом методе пороговое значение и параметр ε адаптируются к данным. Методы Mean-shift [4] для оценки локальной плотности точек используют ядерные функции, построенных по типу функций Гаусса. Эти методы позволяют выделять моды кластеров. Большинство подобных методов используют некоторую метрику расстояния между образами или ее оценку. Как правило это евклидово расстояние, косинусная мера или их вариации.

В многомерном пространстве наблюдается эффект концентрации меры [5,6], который проявляется в том, что чувствительность метрик к вариации образов падает, что затрудняет их использование для оценки плотности точек в признаковом пространстве. Этот эффект присутствует и в любом другом алгоритме, основанном на оценке

расстояния. При увеличении размерности пространства наблюдается также эффект экспоненциального роста вычислительной сложности алгоритмов, что получило название «Проклятие размерности».

Выбор информативных признаков класса

В распознающей системе, существует подсистема измерений, которая отображает объекты в образы – точки признакового пространства классификатора. Будем полагать, что такое отображение является однозначным [7]. Однозначность по существу означает отсутствие противоречий в пространстве образов, т.е. при однозначной схеме измерения подмножества объектов отвечающих разным образам не пересекаются.

Информативные признаки класса образуют подпространство в признаковом пространстве. На качественном уровне можно предположить, что признак обладает достаточной информативностью, если при выполнении условия однозначности он «равномерно» изменяет своё значение при переборе всех представителей класса. Условие «равномерности» можно уточнить следующим образом. Будем считать, что признак обладает высокой информативностью для данного класса, если плотность распределения числа точек класса по диапазону изменения признака является равномерной или близка к ней. Эмпирическая плотность распределения вычисляется как гистограмма числа образов класса на интервальном разбиении диапазона изменения признака. Для информативного признака каждому интервалу дискретности («бину» гистограммы) должны соответствовать примерно равные доли числа образов класса. Для оценки степени равномерности гистограммы можно использовать критерий энтропии:

$$S = - \sum_{i=1}^r \frac{n_i}{N} \log \frac{n_i}{N}.$$

В этом выражении N – число образов в классе, n_i – число образов попадающих в i -й интервал диапазона изменения признака, r – число интервалов. Если все n_i одинаковы и равны N/r , то энтропия принимает максимальное значение, равное $\log(r)$. В качестве критерия информативности удобно использовать нормированную величину $S/\log(r)$ с диапазоном значений от 0 до 1. Ноль будет соответствовать признаку с минимальной информативностью, а единица – с максимальной. Выбор числа бинов гистограммы не формализован, известно ряд эвристических правил [8], позволяющих выбрать это число в автоматическом режиме. При поиске наиболее информативных признаков класса из полного признакового пространства выбираются признаки, которые имеют значение нормированной энтропии гистограммы выше порогового уровня. Экспериментально получено, что значение нормированной энтропии равное 0.75 является достаточным уровнем информативности признака. Гистограммы, соответствующие наиболее информативным признакам, назовём координатными профилями класса.

Координатные профили

При выполнении гипотезы компактности для кластеров, можно ожидать, что вид координатного профиля будет в той или иной мере отражать кластерную структуру класса.

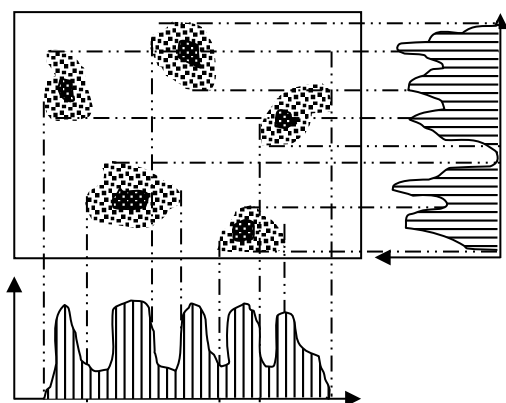


Рис. 1. Координатные профили

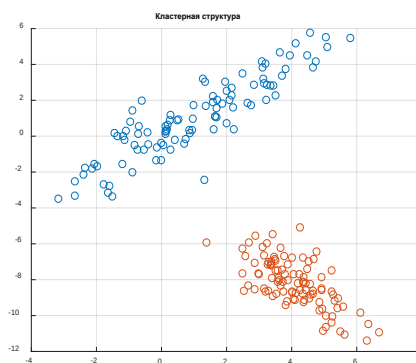


Рис. 2 Пример кластерной структуры

Число мод на каждой координатной гистограмме (см. рис. 1), очевидно, не может превысить числа кластеров. Точки с координатами, принадлежащими малым окрестностям какой-либо моды, потенциально являются кандидатами точек, принадлежащих кластерам класса. При построении профильных гистограмм зафиксируем также принадлежность точек класса бинам каждого координатного профиля и назовём эти данные *профильными векторами принадлежности точек*.

Кластерные зоны

На координатном профиле нанесём линию порогового уровня и выделим области (зоны), в которых плотность точек превышает пороговый уровень. На рис. 3 эти области показаны штриховкой на горизонтальной оси. Каждой зоне присвоим числовую метку. Число построенных зон является оценкой числа кластеров класса. Эта величина, очевидно, зависит от позиции линии порогового уровня. Если пороговый уровень близок к нулю, то число зон уменьшается до единицы, и мы теряем информацию о кластерной структуре, при увеличении порогового уровня число кластерных зон возрастает до максимального, но при этом снижается покрытие кластерными зонами точек класса, и мы рискуем получить кластерную структуру с низким уровнем представительности.

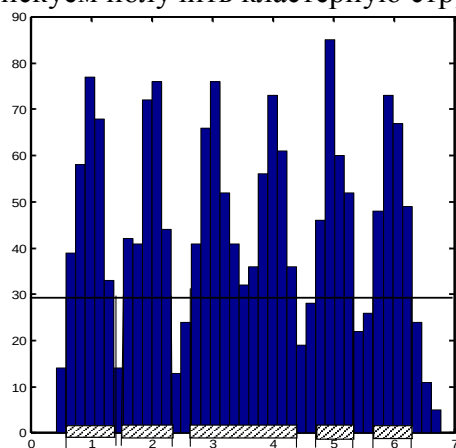


Рис. 3. Кластерные зоны координатного профиля

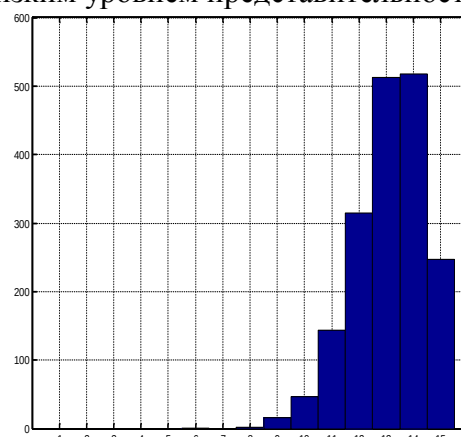


Рис. 4. Гистограмма видимости точек по кластерным зонам профилей

Для решения задачи выбора кластерных зон профиля, будем перемещать линию уровня с шагом 1 от нулевого до максимального значения гистограммы, и при каждом положении порогового уровня вычислять число кластерных зон и суммарный объем их покрытий по точкам класса. В результате получим зависимости числа кластерных зон и их покрытий от значения порогового уровня. На рис. 5 для двух координатных профилей

показаны примеры построения подобных зависимостей для кластерной структуры, показанной на рис. .

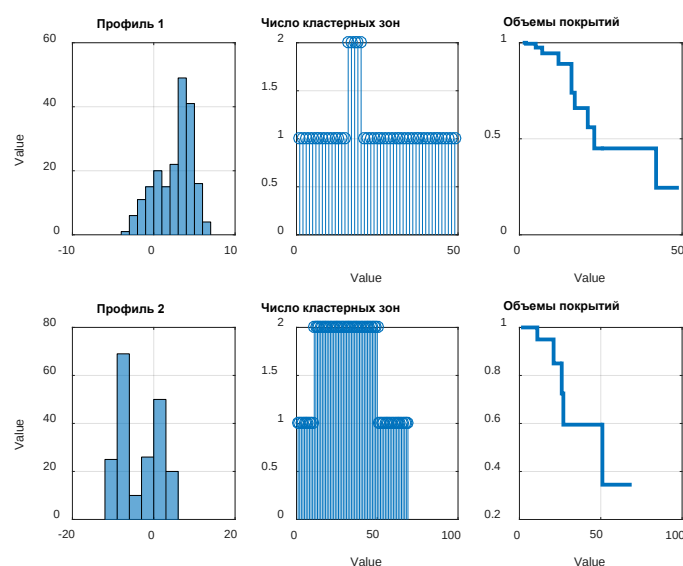


Рис. 5 Алгоритм покоординатной локализации кластерных зон

На данном рисунке координатная ось *Value* означает пороговый уровень координатного профиля. Объёмы покрытий нормированы по отношению к полному числу точек класса. Функция объёма покрытия является монотонно убывающей по значению *Value*. Функция числа кластерных зон состоит из участков с постоянным значением. Очевидно, что это справедливо для любых профилей, поэтому значения *Value* в начале каждого участка постоянства функции числа кластерных зон, являются подходящим выбором порогового уровня, поскольку кластерные зоны в этом случае имеют наибольшее суммарное покрытие.

Протяжённость участков постоянства функции в терминах топологического анализа данных называется персистентностью [9] или устойчивостью топологической характеристики координатного профиля. При выборе порогового уровня *Value* из числа подходящих можно руководствоваться тремя критериям: персистентностью, числом кластерных зон и объёмом покрытия в начале участка постоянства.

Алгоритм редукционной кластеризации

Предположим, что из числа подходящих значений пороговых уровней, выбрано наилучшим образом единственное значение *Value* для каждого профиля. По выбранным значениям *Value* построим кластерные зоны профилей и присвоим зонам индивидуальные локальные метки. Используя построенные ранее профильные векторы принадлежности, для каждого координатного профиля пометим все точки признакового пространства класса, метками кластерных зон. В отдельных профилях некоторые точки могут остаться не помеченными, если они не принадлежат ни одной кластерной зоне. Результаты соберём в таблицу (см. табл. 1).

Табл. 1. Распределение образов по кластерным зонам координатных профилей

Номер образа	Метки кластерных зон				
	Профиль 1	Профиль 2	...	Профиль m-1	Профиль m
1					
2					
...	...	...	...	...	...
N					

В этой таблице первый столбец содержит порядковые номера образов класса, а остальные столбцы заполнены метками кластерных зон по координатным профилям. Некоторые клетки таблицы могут оказаться пустыми. Если строка образа полностью заполнена метками зон, то данный образ проявляется в кластерных зонах всех координатных профилей и это означает, что он принадлежит некоторому кластеру в полном признаковом подпространстве класса. Группируя все строки таблицы с выбранным набором меток, получим состав данного кластера. Строки, заполненные другим набором меток, формируют наполнение следующего кластера и т.д. Таким образом, в результате селекции уникальных строк таблицы будет полностью построена кластерная структура класса.

Наглядной характеристикой наполненности таблицы является гистограмма, представленная на рис. 4 (гистограмма построена для 15-мерного признакового пространства). Горизонтальная ось гистограммы определяет количество заполненных клеток в строке таблицы, а вертикальная ось – количество строк таблицы с одинаковой степенью наполненности. Гистограмма показывает количество точек класса, которые видны через координатные зоны в заданном числе профилей, поэтому эту гистограмму можно назвать *гистограммой видимости кластеров* или *гистограммой координатной встречаемости* точек [10]. Гистограмма количественно характеризует объемы покрытий кластеров различной размерности в признаковом пространстве класса. Последний столбец гистограммы соответствует количеству строк таблицы с полной наполненностью по строке, и определяет объем покрытия для полноразмерной кластерной структуры. Объем покрытия полноразмерной кластерной структуры существенно меньше размера выборки т.е. происходит редукция числа точек класса при сохранении кластерных центров. Поэтому данный метод локализации кластерных центров уместно назвать редукционным. Для Больших данных это свойство алгоритма может играть положительную роль, поскольку редукция делает Большие данные обозримыми без потери информации о кластерных центрах [11]

Оценкой качества кластеризации может служить суммарный объем покрытия кластерной структурой точек класса. Чем он выше, тем лучше качество кластеризации. Максимальное значения данного критерия, очевидно, не превышает число образов в классе.

Следует заметить, что алгоритм редукционной кластеризации не использует какую-либо меру близости в признаковом пространстве, поэтому он применим к выборкам данных представленных в любой шкале от интервальной до номинальной.

Кластерный анализ

Цель кластерного анализа заключается в том, чтобы определить все возможные полноразмерные кластерные структуры класса и установить соотношения между ними. Для решения данной задачи просматриваются все комбинации подходящих пороговых значений профильных зон, для каждой комбинации выполняется редукционная кластеризация и оценка покрытия найденной кластерной структуры. При больших размерностях признакового пространства, прямой перебор может потребовать значительных вычислительных затрат. Для сокращения объёма вычислений, можно ввести дополнительные ограничения, например, установить минимально допустимый размер покрытия для кластерной структуры. В этом случае в процедуре перебора участвуют только те уровневые значения профильных зон, покрытия которых не меньше

чем заданное покрытие кластерной структуры. Результаты полного кластерного анализа используются для детальной оценки конфигурации класса.

Для решения задачи классификации в большинстве случаев достаточно построить в каждом классе только одну кластерную структуру. Эта структура должна быть представительна по покрытию и иметь по возможности максимальное число кластеров. Идеальным решением является первоначальное построение всех допустимых кластерных структур класса, а затем выбор из них наилучшей. Но как уже было отмечено этот путь ведёт к существенным вычислительным затратам. Поэтому имеет смысл предложить некоторые эвристические решения, позволяющие сразу найти одну оптимальную структуру или структуру близкую к оптимальной. Эвристики сводятся к оптимальному выбору единственного подходящего распределения кластерных зон на координатных профилях. Параметрами такого выбора являются: объем покрытия кластерными зонами, число кластерных зон и протяжённость участка постоянства функции числа кластерных зон.

Удовлетворительные результаты можно получить, если размер зонового покрытия установить, как ограничение (например, больше чем 0.75) и выбрать при этом пороговый уровень профиля с максимальным числом кластерных зон или с максимальной протяжённостью участка постоянства функции числа кластерных зон.

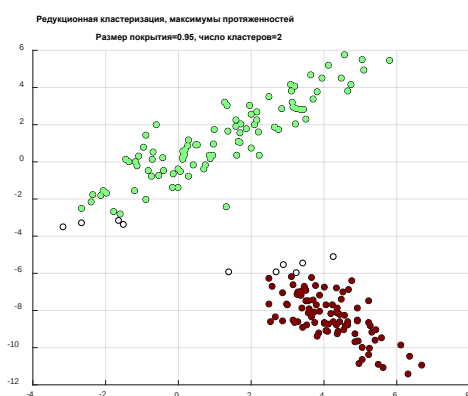


Рис. 6. Выбор по максимуму протяжённости участков постоянства в профилях. Кластерная структура 1

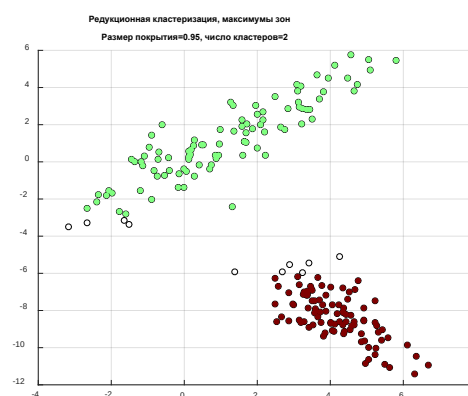


Рис. 7. Выбор по максимуму числа кластерных зон в профилях. Кластерная структура 2

На рис. 6 и рис. 7 показаны кластерные структуры, построенные с использованием двух эвристик для двумерного признакового пространства. Не кластеризованные точки на рисунках представлены не закрашенными маркерами. Из рисунков видно, что результаты кластеризации по обеим эвристикам совпали. Моды кластеров легко определяются усреднением образов в пределах кластера. Совокупность мод кластеров составляет набор эталонов класса.

Пост кластеризация

Задача пост кластеризации состоит в том, чтобы дополнить кластеры точками класса, которые не попали в построенную кластерную структуру. Наиболее просто это сделать, если ввести функцию расстояния между точкой класса и кластером:

$$L(a, C_k) = \min_i (d(a, c_{ki})),$$

где $C_k = \{c_{ki}\}$ – множество точек определяющих кластер, a – внешняя точка по отношению к кластерной структуре, $d(a, c_{ki})$ – некоторая метрика расстояния между точками. В алгоритме пост кластеризации просматриваются все внешние точки и относят их к ближайшему кластеру.

Возможны два варианта реализации этого алгоритма. В первом – просчитывают все расстояния внешних точек от кластеров, определяют ближайший кластер к каждой точке, и затем одновременно дополняют кластеры внешними точками. Во втором варианте дополнения кластеров происходит последовательно по каждой внешней точке, в этом случае на каждом шаге производится расчёт расстояния к уже обновлённым кластерам на предыдущем шаге. На рис. 8 и рис. 9 показаны применения алгоритма пост кластеризации к построенным кластерным структурам. Для рассмотренного примера оба варианта алгоритма дали одинаковый результат.

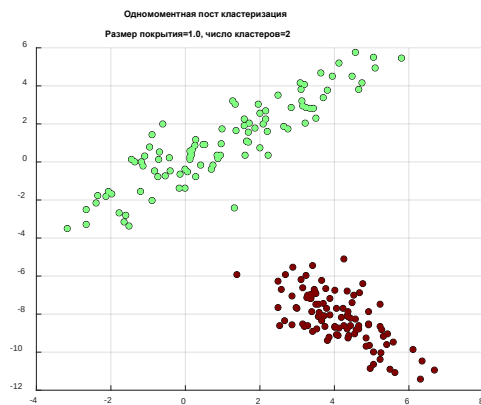


Рис. 8. Результат пост кластеризации.
Кластерная структура 1

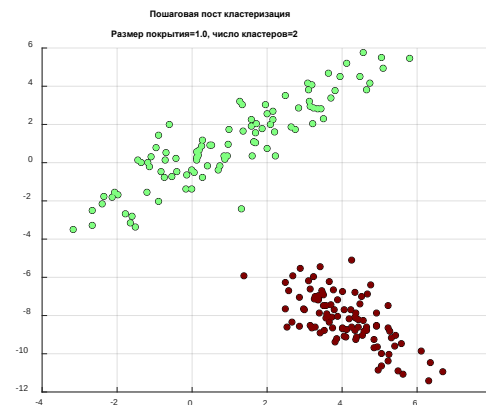


Рис. 9. Результат пост кластеризации.
Кластерная структура 2

Классификация образов

Функцию расстояния между точкой и множеством можно использовать также для классификации образов. В этом случае принадлежность точки классу определяется по минимальному расстоянию между распознаваемой точкой и кластерами классов. Для Больших данных такой подход неэффективен по вычислительным затратам, поскольку требует оценки расстояния между распознаваемой точкой и всеми точками обучающей выборки. Более эффективным решением является сравнения распознаваемой точки с эталонами класса. При использовании эталонов подходящей мерой принадлежности может служить линейный дискриминант [12] для пары векторов «образ, эталон».

Заключение

Метод редукционной кластеризации эффективно решает задачу построения кластерных структур для данных высокой размерности. Результатом являются полноразмерные кластерные зоны, позволяющие построить эталоны классов и выполнить посткластеризацию для нераспределенных образов. Алгоритм редукционной кластеризации может быть использован как для количественных, так и для

категориальных данных. Выделение кластерной структуры классов упрощает построение надежных классификаторов образов в высокоразмерном признаковом пространстве. Метод может быть развит для построения кластерных структур на подпространствах признаков пространств классов.

Список литературы

1. Ту Дж., Гонсалес Р. Принципы распознавания образов // М. 1978. 411с.
2. Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg; Xu, Xiaowei (1996). Simoudis, Evangelos; Han, Jiawei; Fayyad, Usama M. (eds.). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). AAAI Press. pp. 226–231.
3. Ankerst, Mihael; Breunig, Markus M.; Kriegel, Hans-Peter; Sander, Jörg (1999). "OPTICS: Ordering Points To Identify the Clustering Structure". ACM SIGMOD international conference on Management of data. ACM Press. pp. 49–60.
4. Achtert, E.; Böhm, C.; Kröger, P. (2006). "DeLi-Clu: Boosting Robustness, Completeness, Usability, and Efficiency of Hierarchical Clustering by a Closest Pair Ranking". Advances in Knowledge Discovery and Data Mining. Lecture Notes in Computer Science. 3918. pp. 119–128
5. Зорич В.А. Многомерная геометрия, функции очень многих переменных и вероятность // Теория вероятн. и ее примен., 2014, том 59, выпуск 3, С. 436–451/ DOI:<https://doi.org/10.4213/tvp4578>
6. Мильман В.Д. Явления, возникающие в высоких размерностях.— Успехи матем. наук, 2004, т.59, №1, с.157–168.
7. Гисин В. Б., Цаленко М. Ш. Алгебраическая теория систем и её приложения. Системные исследования. Методологические проблемы: Ежегодник // 1984. М.: Наука, 1984. С. 130–151.
8. Histogram From Wikipedia URL:https://en.wikipedia.org/wiki/Histogram#cite_note-13. (дата обращения: 30.05.2020).
9. Макаренко Н.Г., Князева И.С. Топология и данные. Труды XX международной научно-технической конференции «Нейроинформатика-2018». Лекции по нейроинформатике. М.: НИЯУ МИФИ, 2018. – С. 72-108.
10. Дорогов А.Ю, Курбанов Р.Г. Шестопалов М.Ю. Редукция выборки в задачах многомерной кластеризации // Сборник докладов Всероссийской научной конференции Управление и информационные технологии (УИТ-2005). Санкт-Петербург 30 июня-2 июля 2005. Т.1.– С. 284-288.
11. Дорогов А.Ю. Проблемы и вызовы больших данных в нейротехнологии // XVIII Международная научно-техническая конференция «Нейроинформатика-2016»: Лекции по нейроинформатике. М.: НИЯУ МИФИ, 2016.– С. 46-80.
12. Fisher, R.A. The Use of Multiple Measurements in Taxonomic Problems // Annals of Eugenics. — 1936 Т. 7. — С. 179-188.

КЛАСТЕРИЗАЦИЯ ПОКУПАТЕЛЕЙ СИСТЕМЫ МАКСБОНУС С ПРИМЕНЕНИЕМ САМООРГАНИЗУЮЩИХСЯ КАРТ КОХОНЕНА

М. Г. Доррер

Сибирский государственный университет науки и технологий им. М.Ф. Решетнева,

dorrer_mg@sibsau.ru

А.В. Фомин, Д.А. Логинов

ООО «Фабрика решений»,

a.v.fomin@live.com

Введение

Данная работа выполнена в рамках проекта «Разработка интеллектуальной системы анализа и управления программой лояльности – «Максбонус» [1], выполняющегося при поддержке Фонда содействия инновациям (Foundation for Assistance to Small Innovative Enterprises, FASIE) [2] в рамках договора о предоставлении гранта на проведение научно-исследовательских и опытно-конструкторских работ в рамках проекта.

Система «МаксБонус» реализуется для оперативного уведомления владельца бизнеса об отклонениях в основных бизнес-процессах обслуживания клиентов и рекомендациях, связанных с этим. Также целью создания системы является консультативная поддержка владельца бизнеса относительно основных показателей программы лояльности и клиентского сервиса, информирование относительно основных возможностей программы лояльности и автоматическое создание настроек соответствующих процессов и акций.

Решению задач исследованию потребительских предпочтений с использованием больших данных посвящено достаточно большое количество научных работ. Статьи [3], [4], [5] посвящены анализу потребительских предпочтений при покупках онлайн. Aichner и Coletti рассматривают [6] процесс переориентации потребителей при онлайн покупок с массовых продуктов на персонализированные. В работах [7], [8] авторы анализируют кластеризацию покупателей на основе потребительских предложений. Авторы данного доклада также предпринимали попытки анализа потребительских предпочтений при помощи алгоритмов анализа ассоциативных правил [9].

Таким образом, задача кластеризации потребителей по их предпочтениям актуальна и востребована. При этом не удалось встретить статей, описывающих применение для решения этой задачи искусственных нейронных сетей без учителя. Итак, данная работа посвящена задаче кластеризации потребителей торговой сети на основании данных системы потребительской лояльности при помощи самоорганизующихся карт Кохонена.

Исходные данные и методы исследования

Система «МаксБонус» спроектирована таким образом, что в качестве внешних данных может быть использована любая программа лояльности с идентификацией пользователя и состава покупаемых им товаров. Система обладает открытой архитектурой, позволяющей расширять функционал системы за счет добавления новых компонентов.

Компоненты системы:

1. Внешняя программа лояльности
2. Модуль загрузчик
3. Модуль хранения
4. Автоматизированное рабочее место эксперта
5. Модуль взаимодействия

6. Чат-бот пользователя
7. Модуль автоматической фиксации

Проектируемая и реализуемая система использует в качестве внешних источников данных программу лояльности. Содержащиеся в ней информация это оцифрованное покупательское поведение клиентов у конкретного партнера. Весь набор знаний, которые можно получить из этих данных представляет собой результат анализа покупательского поведения клиентов. В результате анализа потребностей партнеров командой проекта был отобран и реализован следующий набор факторов, для формирования которых разработаны группы алгоритмов:

- Средний чек
- Количество чеков
- Сумма чеков
- Средний цикл покупки
- Доля оцененных чеков
- Средняя оценка покупки

Алгоритмы сегментации используются для расчета факторов:

- Доля новых клиентов
- Доля клиентов с бесплатными каналами коммуникации
- Доля лояльных клиентов
- Доля потерянных клиентов

Алгоритмы сегментации и факторы, которые создаются в системе по итогам их работы представляют наибольшую ценность для партнера программы лояльности, т.к. получение такой информации альтернативным способом требует от партнера большой аналитической работы. Ожидается, что после полной реализации запланированных функций системы она позволит пользователю через чат-бота получать рекомендации приблизительно следующего содержания:

«Я заметил, что в мае у вас появилось 147 клиентов, которые перестали покупать категорию «мясо» у вас в магазине, они продолжают к вам ходить, но тратят при этом на 24%. меньше. Я рекомендую запустить для этих клиентов персональную акцию на данную категорию товаров, чтобы вернуть их к первоначальному поведению».

Результаты загрузки данных системы лояльности модулем загрузчика позволяют провести автоматическую кластеризацию/сегментацию различных субъектов анализа потребительского поведения. Для решения задачи кластеризации применяется нейросетевая кластеризации без учителя при помощи самоорганизующихся карт Кохонена [10].

Прототипирование алгоритмов кластеризации производилось в нейросетевом модуле Deep Learning Toolbox [11] системы Matlab с последующим переносом в ядро системы. Использовалась функция `selforgmap`, реализующая модель самоорганизующейся карты Кохонена с миграцией данных из системы МаксБонус через файлы Excel. На Рис. 1 приведена архитектура карты Кохонена, используемой для кластеризации покупателей одного из партнеров программы на 6 классов по 11 признакам.

Self-Organizing Map (view)

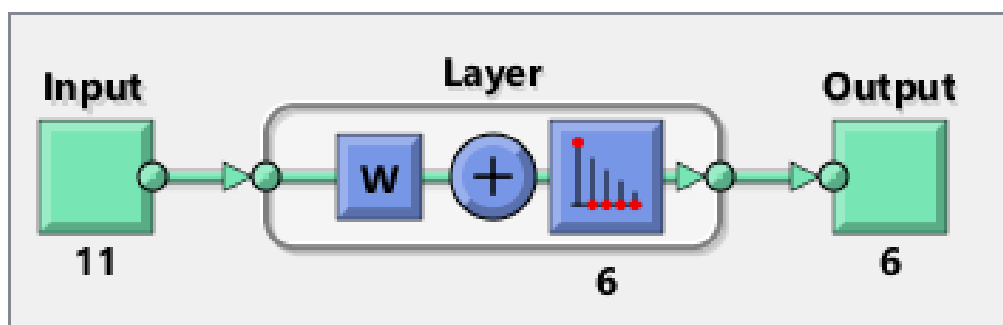


Рис. 1. Архитектура самоорганизующейся сети Кохонена для кластеризации клиентов

Результаты кластеризации

На Рисунке 2 показано распределение 990 покупателей по 6 классам. Видно, что большинство покупателей сеть сгруппировала в один крупный класс. Были проведены эксперименты для числа классов от 2 до 8, подобная картина разбиения повторялась постоянно, сеть выделяла два класса, повторяющиеся при любом разбиении, остальные классы варьировались. При разбиении более, чем на 4 класса сеть начала выделять «пустые» классы, как это видно на Рис. 2.

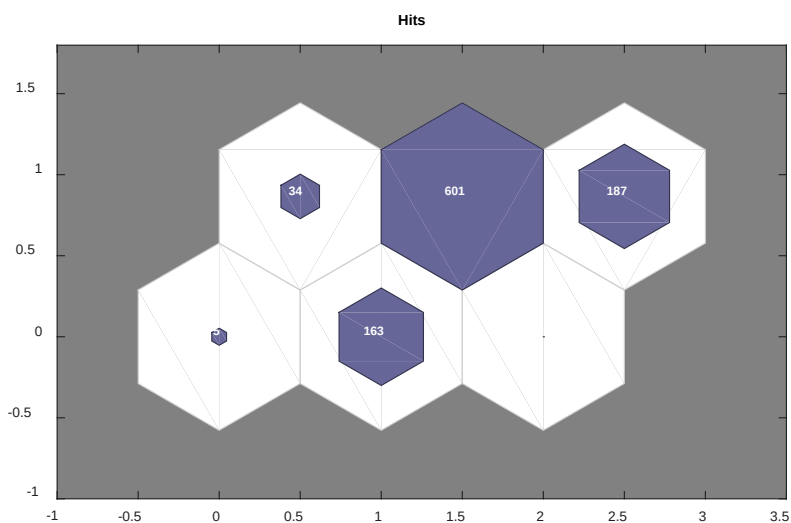


Рис. 2. Распределение покупателей по 6 классам

В процессе исследования проводилась оценка стабильности классов при кластеризации, разделения покупателей по классам в процессе наращивания числа классов.

Одна и та же выборка (покупатели сети мясных магазинов) подвергались автоматической кластеризации сетью Кохонена на количество классов от 2 до 6. Далее производилась оценка совместного попадания одного и того же покупателя в N классов и в $N+1$ класс для $N=1..5$.

В Таблицах 1 – 4 показано перераспределение покупателей между классами с ростом числа классов. По строкам перечислены классы меньшего подмножества (на N классов), по столбцам – большего (на $N+1$ класс). На пересечении – число покупателей, попавших в оба класса.

Таблица 1

Пересечение между классами (кластеризация на 2 и на 3 класса)

Количество покупателей	Распределение по 3 классам			
Распределение по 2 классам	1	2	3	Общий итог
1	19	20		39
2		117	834	951
Общий итог	19	137	834	990

Таблица 2

Пересечение между классами (кластеризация на 3 и на 4 класса)

Количество покупателей	Распределение по 4 классам				
Распределение по 3 классам	1	2	3	4	Общий итог
1	19				19
2		20	117		137
3		647		187	834
Общий итог	19	667	117	187	990

Таблица 3

Пересечение между классами (кластеризация на 4 и на 5 классов)

Количество покупателей	Распределение по 5 классам					
Распределение по 4 классам	1	2	3	4	5	Общий итог
1	5	14				19
2			66	601		667
3		20	97			117
4					187	187
Общий итог	5	34	163	601	187	990

Таблица 4

Пересечение между классами (кластеризация на 5 и на 6 классов)

Количество покупателей	Распределение по 6 классам					
Распределение по 5 классам	1	2	4	5	6	Общий итог
1	5					5
2			34			34
3		151		12		163
4				601		601
5					187	187
Общий итог	5	151	34	613	187	990

В таблицах 5 – 9 представлены усредненные характеристики классов. Усреднение велось за весь период работы исследуемой торговой сети, в перспективе алгоритм будет настроен на усреднение за типовой диапазон времени – месяц, квартал, год. При усреднении использовались следующие поля базы данных:

- Gender – пол покупателя;
- Age – возраст покупателя;
- Sum_amount – общая сумма покупок за период;
- Check_count – количество чеков за период;
- Avg_amount – средний чек за период.

Таблица 5

Усредненные характеристики классов при распределении на 2 класса

Класс	Количество покупателей в классе	Среднее по полю gender	Среднее по полю age	Среднее по полю sum_amount	Среднее по полю check_count	Среднее по полю avg_amount
1	39	0,38	39,18	52866,87	156,49	415,49
2	951	0,32	43,55	4421,17	12,45	431,52
Итого	990	0,32	43,38	6329,64	18,12	430,89

Таблица 6

Усредненные характеристики классов при распределении на 3 класса

Класс	Количество покупателей в классе	Среднее по полю gender	Среднее по полю age	Среднее по полю sum_amount	Среднее по полю check_count	Среднее по полю avg_amount
1	19	0,37	38,74	70987,26	217,11	411,98
2	137	0,36	43,33	19111,29	50,26	502,09
3	834	0,31	43,49	2757,00	8,31	419,62
Итого	990	0,32	43,38	6329,64	18,12	430,89

Таблица 7

Усредненные характеристики классов при распределении на 4 класса

Класс	Количество покупателей в классе	Среднее по полю gender	Среднее по полю age	Среднее по полю sum_amount	Среднее по полю check_count	Среднее по полю avg_amount
1	19	0,37	38,74	70987,26	217,11	411,98
2	667	0,31	43,81	3640,30	11,10	395,08
3	117	0,33	43,92	20442,30	53,15	516,38
4	187	0,35	41,97	522,75	1,06	507,06
Итого	990	0,32	43,38	6329,64	18,12	430,89

Таблица 8

Усредненные характеристики классов при распределении на 5 классов

Класс	Количество покупателей в классе	Среднее по полу gender	Среднее по полу age	Среднее по полу sum_amount	Среднее по полу check_count	Среднее по полу avg_amount
1	5	0,60	50,80	109920,11	416,60	285,72
2	34	0,35	37,47	44476,69	118,24	434,57
3	163	0,32	44,59	14361,78	37,80	496,73
4	601	0,31	43,76	2938,12	9,12	390,33
5	187	0,35	41,97	522,75	1,06	507,06
Итог	990	0,32	43,38	6329,64	18,12	430,89

Таблица 9

Усредненные характеристики классов при распределении на 6 классов

Класс	Количество покупателей в классе	Среднее по полу gender	Среднее по полу age	Среднее по полу sum_amount	Среднее по полу check_count	Среднее по полу avg_amount
1	5	0,60	50,80	109920,11	416,60	285,72
2	151	0,34	44,59	14803,08	38,63	508,45
4	34	0,35	37,47	44476,69	118,24	434,57
5	613	0,30	43,77	3053,04	9,48	389,53
6	187	0,35	41,97	522,75	1,06	507,06
Итог	990	0,32	43,38	6329,64	18,12	430,89

Обсуждение результатов

Даже не анализируя распределение параметров покупателей по классам, можно сделать вывод, что выборка содержит значимое стабильное ядро экземпляров (600 – 640 покупателей). Этот ядерный класс (при 4 классах – второй, при 5 классах – четвертый, при 6 классах – пятый) не начинает дробиться при увеличении числа классов, то есть входящие в него покупатели отличаются значимым сходством (линейной близостью) в пространстве признаков.

Анализ параметров покупателей по классам позволяет сделать следующие выводы:

- В «ядерный» класс выделились покупатели с количеством чеков около 10 и средним чеком в 390-400 рублей
- Начиная с 5 классов впервые устойчиво выделяется класс, включающий больше мужчин чем женщин, причем эта категория характеризуется очень большим числом чеков – более 400.
- Четко выделяются в отдельный класс покупатели с 1 чеком – это 187 покупателей, класс выделяется начиная с разбиения на 4 класса и сохраняется неизменным при последующих разбиениях.

Оценивая перспективы данной работы, представляется интересным дополнить проведенный анализ рассмотрением параметров покупок покупателей по товарным группам. Использование таких данных позволит проанализировать, каким образом изменится кластеризация с учетом покупок по товарным группам и получить представление о товарных предпочтениях покупателей ранее выделенных классов.

Также перспективным выглядит проведение кластеризации не только клиентов, но и партнеров. Кластеризация партнеров позволит решить следующие задачи:

- Более обоснованно оценить группировку партнеров по классам, в том числе оценить количество классов, обеспечивающих адекватное разбиение при имеющихся данных.
- Отнести вновь подключившихся к программе лояльности партнеров к одному из ранее заданных классов, что обеспечит формирование рекомендаций для них исходя из накопленного опыта системы.

Список литературы

1. Maxbonus [Electronic resource]. URL: <http://maxbonus.ru/>.
2. Foundation for Assistance to Small Innovative Enterprises (FASIE) [Electronic resource]. URL: <http://fasie.ru/>.
3. Vithayathil J., Dadgar M., Osiri J.K. Social media use and consumer shopping preferences // *Int. J. Inf. Manage.* 2020. Vol. 54. P. 102117.
4. Chen H.-J. What drives consumers' mobile shopping? 4Ps or shopping preferences? // *Asia Pacific J. Mark. Logist.* 2018. Vol. 30, № 4. P. 797–815.
5. Rajamma R.K., Neeley C.R. Antecedents to Shopping Online: A Shopping Preference Perspective // *J. Internet Commer.* 2005. Vol. 4, № 1. P. 63–78.
6. Aichner T., Coletti P. Customers' online shopping preferences in mass customization // *J. Direct, Data Digit. Mark. Pract.* 2013. Vol. 15, № 1. P. 20–35.
7. Schüller D., Pekárek J. Customer Satisfaction Measurement – Clustering Approach // *Acta Univ. Agric. Silvic. Mendelianae Brun.* 2018. Vol. 66, № 2. P. 561–569.
8. Zhang J. et al. An effective collaborative filtering algorithm based on user preference clustering // *Appl. Intell.* 2016. Vol. 45, № 2. P. 230–240.
9. Ogurtsov D.A., Dorrer M.G. Application of association rules learning for studying the store history of a large retail chain // *Journal of Physics: Conference Series.* 2019. Vol. 1399, № 3.
10. Kohonen T. The self-organizing map // *Proc. IEEE.* 1990. Vol. 78, № 9. P. 1464–1480.
11. Deep Learning Toolbox [Electronic resource]. URL: <https://www.mathworks.com/help/deeplearning/>.

СРАВНЕНИЕ ЭФФЕКТИВНОСТИ РАБОТЫ АРХИТЕКТУР YOLOV3 И MASC R-CNN В КОМПЬЮТЕРНОМ ЗРЕНИИ УМНОГО ХОЛОДИЛЬНИКА

М. Г. Доррер

Сибирский государственный университет науки и технологий им. М.Ф. Решетнева,
dorrer_mg@sibsau.ru

А. Е. Толмачева

«Фабрика решений», *vkarmeo@mail.ru*

Введение

В настоящее время в задачах классификации объектов в режиме реального времени необходимо чем-то жертвовать. Модель работает медленно, но классифицирует объекты качественно, или же операция распознавания выполняется быстро, но качество определения резко падает.

Для каждой задачи необходимо сделать данный выбор и подобрать оптимальную архитектуру.

Например, в статье [1] Medina et al. решают проблему адекватной классификации изображений в определенные контрольные группы с помощью трансферного обучения с пятью архитектурами CNN, предварительно обученными на Imagenet, чтобы определить, какая из них лучше всего классифицирует скриншоты, полученные из Industrial Controls Systems.

Авторы работы [2] решают проблему распознавания изображений в условиях различных типов шумов разметки: ложноотрицательные, ложные срабатывания и неточные границы. Для решения этой проблемы они предложили новый детектор объектов NOise Tolerant Ensemble RCNN (NOTE-RCNN), обеспечивающий обработку зашумленных меток.

В работе [3] Bergman et al. использовали методы поиска нейронной архитектуры (NAS) и MRF для подбора оптимальной архитектуры модели, подходящей для выполнения текущих задач.

В статье [4] предлагается метод обнаружения трехмерных объектов, основанный на облаке точек и изображении.

Сочетание архитектуры YOLO с применением метода RCNN также нередко применяется на практике. Так, в статье [5] описывается решение задачи оценки траектории теннисного мяча на основе захвата видеопотока. В работе [6] автор исследует сравнительную эффективность архитектуры YOLO с существующими методами анализа видеопотока, в том числе показывает преимущества YOLOv3 над R-CNN на целом ряде задач.

Авторы данного доклада также сталкивались в своей работе с применением как методов R-CNN, так и с архитектурой YOLO различных версий. Так, в работе [7] метод Mask-CNN использовался для детекции клеток на микрофотографиях. В работе [8] рассматривалась задача, подобная рассмотренной в статье [5] – отслеживание шайбы в игре. Работа [9] посвящена сравнительному анализу эффективности аппаратных платформ для задач видеодетекции при помощи решений на базе архитектуры YOLO.

Таким образом, как обзор выполненных работ, так и практический опыт авторов подтверждают актуальность задачи сравнения эффективности применения архитектуры

YOLO и метода Mask-CNN для детекции объектов в реальном времени на основании данных видеопотока. Сравнение производилось на данных системы искусственного зрения «умного холодильника», реализованного в рамках проекта «РобиМаркет» [10].

Использованные методы и архитектуры

YOLO или You Only Look Once [11] — это очень популярная на текущий момент архитектура CNN, которая используется для распознавания множественных объектов на изображении.

На рис. 1 показана архитектура YOLOv3

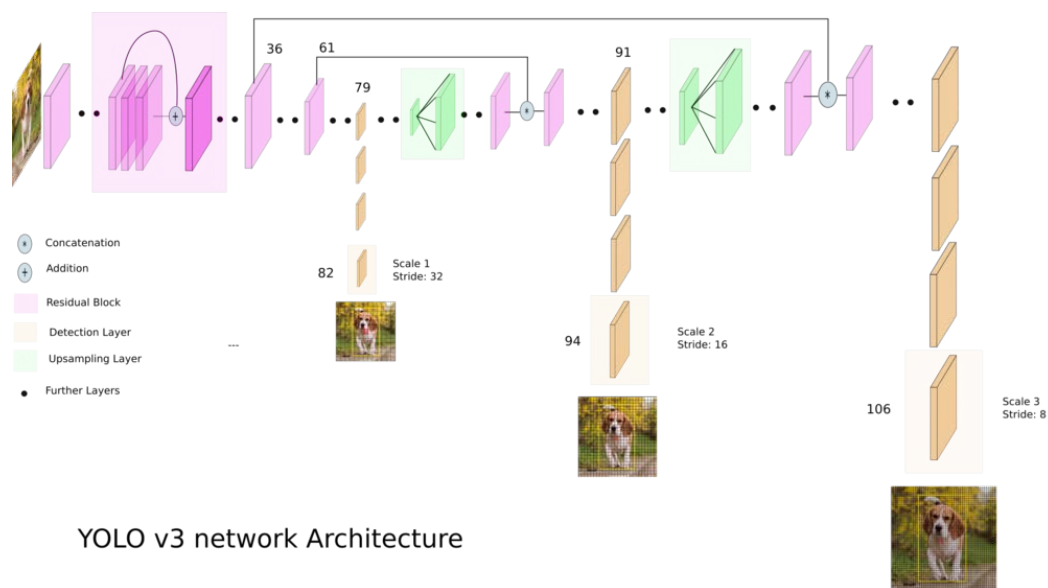


Рис. 1. Визуальный граф архитектуры YOLOv3

Плюсы данного подхода состоят в том, что сеть смотрит на все изображение сразу и учитывает контекст при детектировании и распознавании объекта. Так же YOLO в 1000 раз быстрее чем R-CNN и около 100x быстрее чем Fast R-CNN.

YOLOv3 — это усовершенствованная версия архитектуры YOLO. Она состоит из 106-ти свёрточных слоев и лучше детектирует небольшие объекты по сравнению с её предшественницей YOLOv2. Основная особенность YOLOv3 состоит в том, что на выходе есть три слоя каждый из которых рассчитан на обнаружения объектов разного размера.

Концепции, лежащие в основе в Mask R-CNN [12], прошли поэтапное развитие через архитектуры нескольких промежуточных нейросетей, решавших разные задачи из приведённого выше списка. Рассмотрим последовательно эти этапы.

Mask R-CNN развивает архитектуру Faster R-CNN [13] путём добавления ещё одной ветки, которая предсказывает положение маски, покрывающей найденный объект, и, таким образом решает уже задачу instance segmentation. Маска представляет собой прямоугольную матрицу, в которой «1» на некоторой позиции означает принадлежность соответствующего пикселя объекту заданного класса, «0» — что пиксель объекту не принадлежит.

На рис. 2 изображена архитектура MaskRCNN

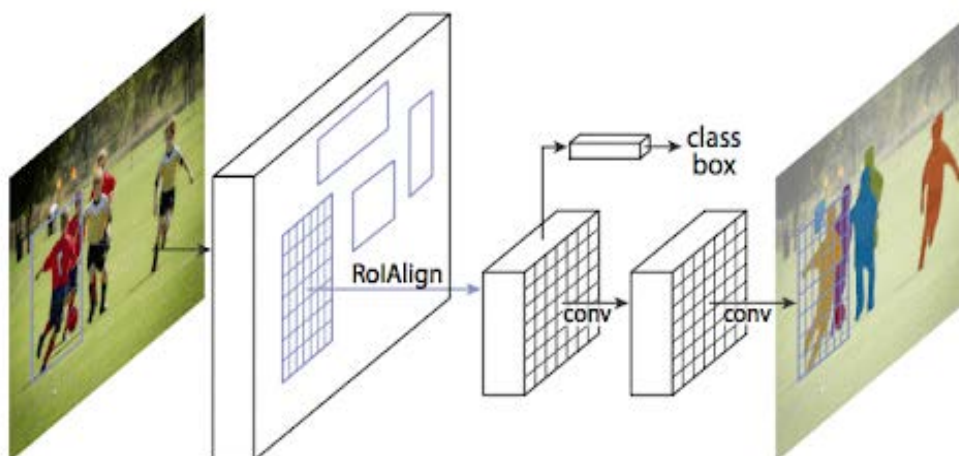


Рис. 2. Архитектура модели Mask RCNN

Авторы подхода условно разделяют разработанную архитектуру на CNN-сеть вычисления признаков изображения, называемую ими backbone, и head — объединение частей, отвечающих за предсказание охватывающей рамки, классификацию объекта и определение его маски.

Выделение маски происходит в class-agnostic стиле: маски предсказываются отдельно для каждого класса, без предварительного знания, что изображено в регионе, и потом просто выбирается маска класса, победившего в независимом классификаторе. Утверждается, что такой подход более эффективен, чем опирающийся на априорное знание класса.

Полученные результаты

На рис. 3 изображен терминал Ubuntu с параметрами нагрузки видеокарты Nvidia 1070 во время этапа обучения архитектуры YOLOv3.

```

nero@nero: ~
nvidia-debugdump      nvidia-persistenced      nvidia-xconfig
nero@nero:~$ nvidia-smi
Fri Jul 5 16:27:11 2019

+-----+
| NVIDIA-SMI 410.48                  Driver Version: 410.48 |
+-----+-----+
| GPU   Name           Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf    Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
+-----+-----+
|  0  GeForce GTX 1070    Off      | 00000000:01:00:0 | On         N/A       |
| 62%   82C   P2      123W / 151W | 3736MiB / 8118MiB | 94%       Default  |
+-----+-----+

+-----+
| Processes:                         GPU Memory |
|   GPU       PID    Type    Process name                     Usage  |
+-----+-----+
|    0         9908    G       /usr/lib/xorg/Xorg                  194MiB |
|    0        10138    G               compiz                      3MiB  |
|    0        10528    G               compiz                      64MiB |
|    0        11847    G       /usr/lib/firefox/firefox            3MiB  |
|    0        14419    C       ./darknet                          3467MiB |
+-----+-----+
nero@nero:~$

```

Рис. 3. Технические показания видеокарты, на котором идет процесс обучения YOLOv3

На рис. 4 изображен терминал Ubuntu с параметрами нагрузки видеокарты Nvidia 1060 во время этапа обучения архитектуры MaskRCNN.

```

deep@deep: ~
(base) deep@deep:~$ nvidia-smi
Wed Jun 24 11:35:17 2020
+-----+
| NVIDIA-SMI 410.48                 Driver Version: 410.48 |
+-----+-----+
| GPU Name Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
+-----+-----+
| 0  GeForce GTX 106...  Off | 00000000:01:00:0  On |          N/A         |
| 44%   33C   P2     29W / 130W | 5865MiB / 6077MiB |      2%    Default   |
+-----+-----+

Processes:
+-----+-----+
| GPU    PID    Type    Process name      | GPU Memory |
|-----+-----+
| 0      1223    G      /usr/lib/xorg/Xorg | 291MiB     |
| 0      2023    G      compiz            | 120MiB     |
| 0      3931    G      ...quest-channel-token=8606249679667201436 | 28MiB      |
| 0      6029    C      python3           | 5413MiB    |
+-----+-----+
(base) deep@deep:~$

```

Рис. 4. Технические показания видеокарты, на которой идет процесс обучения Mask R-CNN.

По нагрузкам видеокарты во время обучения архитектур видно, что архитектура MaskRCNN занимает на 18 % меньше ресурсов, чем архитектура YOLOv3.



Рис. 5. Результат анализа тестового изображения 1

В использованной для экспериментов выборке было 800 изображений, полученных с камер умных холодильников пекарни Тортберри. Изображения были проверены на оригинальность и аугментированность. В тестировании было использовано 70 изображений с встречающимися на них 430 образцами всех классов продукции. В обеих моделях на выходе представлено шесть классов под названиями: red, orange, blue, green, violet, yellow.

Тестирование по показателю скорости решения задачи проводилось на видеокарте Nvidia 1060 на операционной системе Ubuntu 16.04. На рисунках 5 - 7 изображены результаты работы архитектуры YOLOv3.

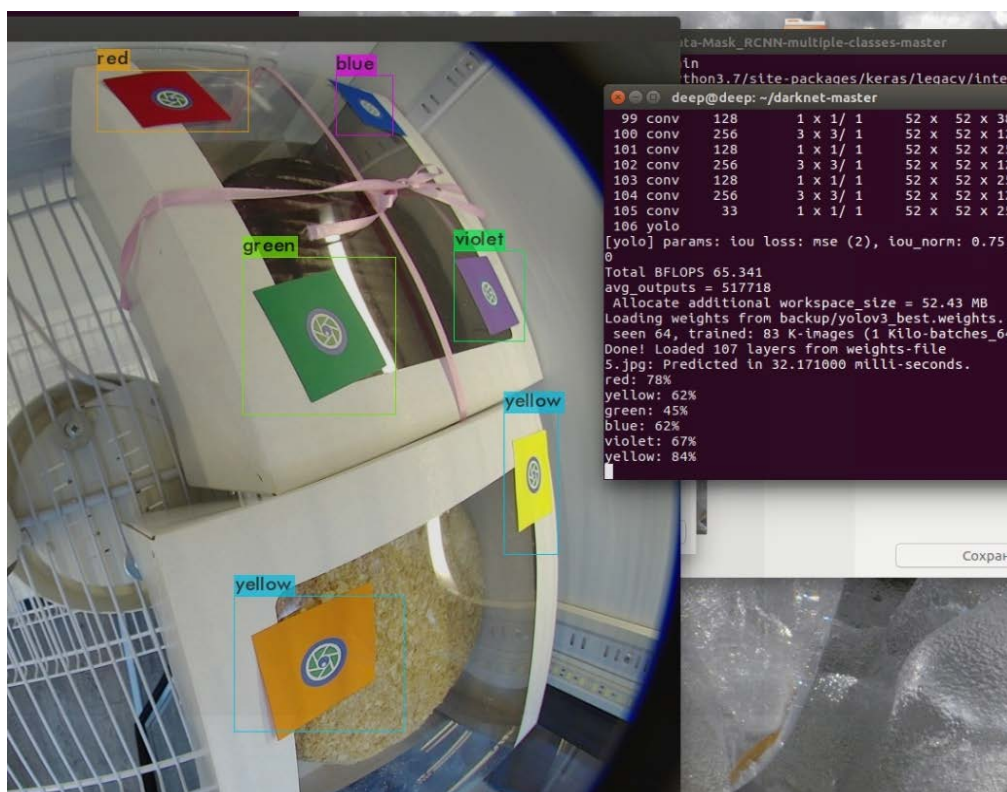


Рис. 6. Результат анализа тестового изображения 2.



Рис. 7. Результат анализа тестового изображения 3.

В таблице 1 приведены результаты сети на каждый образец класса, полученные с тестовых изображений, обработанные моделью архитектуры YOLOv3.

Таблица 1

Результаты анализа модели архитектуры YOLOv3

Номер рисунка	Название класса	Точность%	Скорость обработки видео
Рисунок 5	Red	98	30FPS
	Blue	97	
	Yellow	87	
	Orange	89	
	Green	97	
	Violet	80	
Рисунок 6	Red	78	30FPS
	Blue	62	
	Yellow	62	
	Orange	ошибка	
	Green	45	
	Violet	67	
Рисунок 7	Red	97	30FPS
	Blue	95	
	Yellow	42	
	Orange	ошибка	
	Green	92	
	Violet	74, ошибка	

На рисунках 8 - 10 изображены результаты работы архитектуры Mask R-CNN.



Рис. 8. Результат анализа тестового изображения 1.

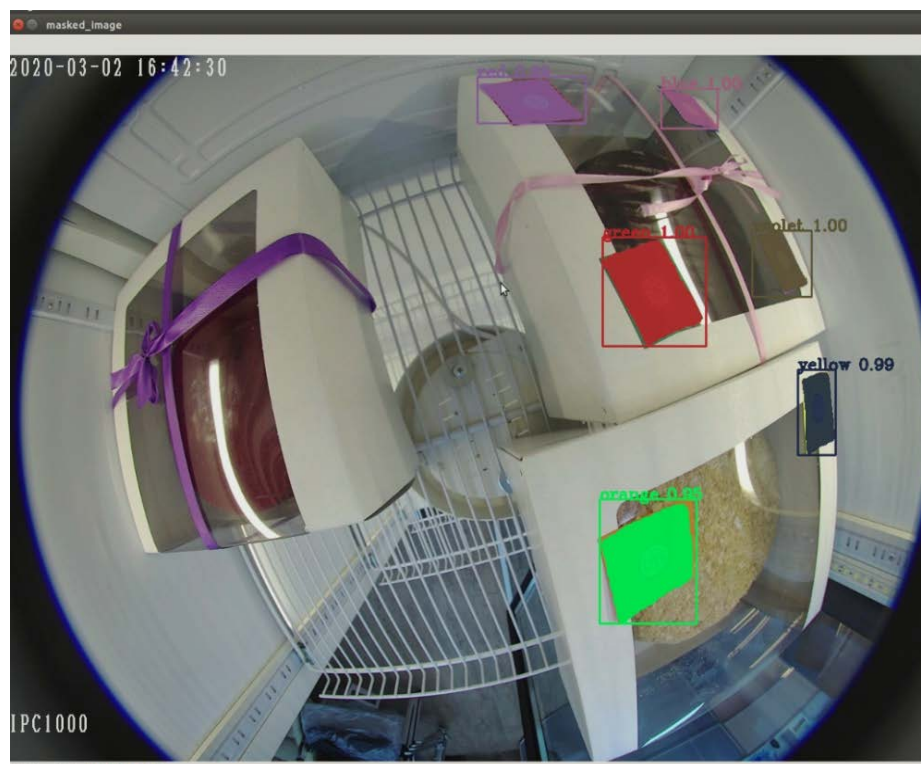


Рис. 9. Результат анализа тестового изображения 2.

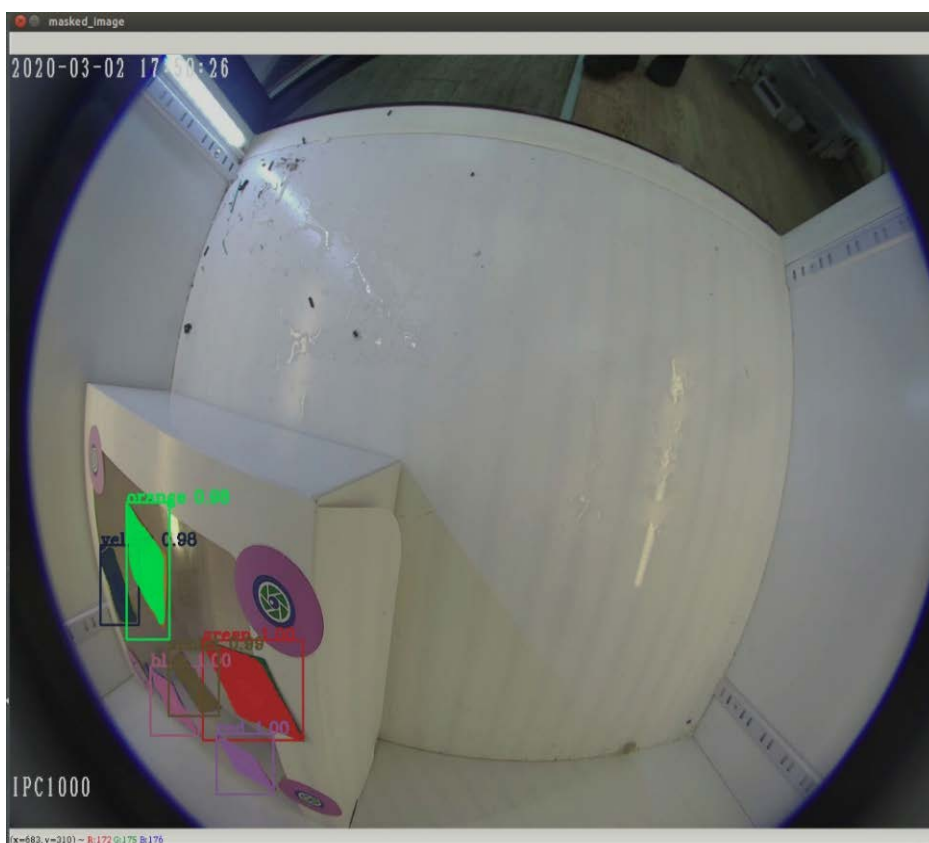


Рис. 10. Результат анализа тестового изображения 3.

В таблице 2 приведены результаты сети на каждый образец класса, полученные с тестовых изображений, обработанные моделью архитектуры Mask R-CNN.

Таблица 2

Результаты анализа модели архитектуры MaskRCNN

Номер рисунка	Название класса	Точность %	Скорость обработки видео
Рисунок 8	Red	99	10FPS
	Blue	99	
	Green	100	
	Violet	98	
Рисунок 9	Red	99	10FPS
	Blue	100	
	Yellow	99	
	Orange	95	
	Green	100	
	Violet	100	
Рисунок 10	Red	100	10FPS
	Blue	100	
	Yellow	98	
	Orange	98	
	Green	100	
	Violet	99	

Обсуждение и выводы

Сравнительный анализ таблиц 1 и 2 показывает, что архитектура Mask R-CNN обладает значительно более высокой точностью определения классов в видеопотоке. Видно, что на одних и тех же классах и на одних и тех же изображениях точность отнесения объектов к классам для Mask R-CNN нигде не падает ниже 95%, а в основном составляет 98-100%. В то же время для архитектуры YOLO в поставленных экспериментах с распознаванием одного из классов вообще не справилась (см. данные таблицы 1), а для нескольких других точность решения задачи составила неприемлемую на практике величину в 42 – 45%.

Вместе с тем, во всех экспериментах производительность на одной и той же аппаратной платформе для архитектуры YOLO оказалась втрое выше, чем для Mask-RCNN.

Кроме того, несмотря на низкую скорость обработки изображения, архитектура Mask R-CNN на тестовых выборках показала высокую точность детектирования каждого образца класса. Данный факт подтверждает помимо высокой точности архитектуры Mask R-CNN еще и достаточность примененного размера обучающей выборки для данной модели.

Таким образом, для использования в системе компьютерного зрения умных холодильников «Робимаркет» [10] перспективным было признано применение архитектуры Mask R-CNN. Ключевым фактором здесь была высокая точность

определения товара, что напрямую связано с коммерческими рисками эксплуатации вендингового автомата. При этом быстродействие на уровне 10FPS, недостаточное для игровых приложений и управления движением в реальном времени, оказалось вполне достаточным для своевременного выставления счета покупателю.

Список литературы

1. Medina P.B. et al. Classification of Industrial Control Systems screenshots using Transfer Learning. 2020.
2. Gao J. et al. NOTE-RCNN: NOise Tolerant Ensemble RCNN for Semi-Supervised Object Detection // 2019 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, 2019. P. 9507–9516.
3. Berman M. et al. AOWS: Adaptive and optimal network width search with latency constraints. 2020.
4. Salomon G., Laroca R., Menotti D. Deep Learning for Image-based Automatic Dial Meter Reading: Dataset and Baselines. 2020.
5. Deepa R. et al. Comparison of Yolo, SSD, Faster RCNN for Real Time Tennis Ball Tracking for Action Decision Networks // 2019 International Conference on Advances in Computing and Communication Engineering (ICACCE). IEEE, 2019. P. 1–4.
6. Du J. Understanding of Object Detection Based on CNN Family and YOLO // J. Phys. Conf. Ser. 2018. Vol. 1004. P. 012029.
7. Yakimov A. et al. The use of convolutional neural networks to identify artifacts of cells micrographs in biomedical research // Journal of Physics: Conference Series. 2019. Vol. 1399, № 3.
8. Tolmacheva A.E., Ogurcov D.A., Dorrer M.G. Puck tracking system for aerohockey game with YOLO2 // Journal of Physics: Conference Series. 2019. Vol. 1399, № 3.
9. Tolmacheva A.E., Ogurtsov D.A., Dorrer M.G. Justification for choosing a single-board hardware computing platform for a neural network performing image processing // IOP Conference Series: Materials Science and Engineering. 2020. Vol. 734, № 1.
10. Robimarket [Electronic resource]. URL: <https://startupvillage.ru/exhibitors/263>.
11. Redmon J. et al. You Only Look Once: Unified, Real-Time Object Detection. 2015.
12. He K. et al. Mask R-CNN. 2017.
13. Ren S. et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. 2015.

НЕЙРОСЕТЕВАЯ КЛАССИФИКАЦИЯ АВИАЦИОННОГО ПЕРСОНАЛА КАК ЭЛЕМЕНТ ИНФОРМАЦИОННО-УПРАВЛЯЮЩЕГО ПРОСТРАНСТВА БЕЗОПАСНОСТИ ОБЪЕКТА ТРАНСПОРТНОЙ ИНФРАСТРУКТУРЫ

Л.Н. Елисов¹, Н.И. Овченков², В.И. Горбаченко³, И.А.Абрамов³

¹ Государственный НИИ гражданской авиации, lev.el@list.ru

² Ярославский государственный университет, ovchenkov@electronika.ru

³ Пензенский государственный университете, gorvi@mail.ru

Авиационная безопасность [1, 2] как научное направление занимается вопросами обеспечения защиты гражданской авиации от незаконного вмешательства в ее деятельность. Целевой функционал системы, решающей эти задачи, по своему содержанию существенно отличается от других систем в составе авиационной транспортной системы, что определяет специфику средств, методов и процедур, применяемых в авиационной безопасности.

На протяжении всей истории возникновения и развития гражданской авиации существует проблема человеческого фактора. Человеческий фактор сегодня является причиной более 80% авиационных происшествий и охватывает практически все стороны производственной деятельности в гражданской авиации [3]. Современная концепция управления авиационной безопасностью авиационной инфраструктуры включает задачу управления авиационным персоналом. Важной задачей управления персоналом авиационной безопасности являются периодически проводимые оценки уровня готовности персонала к выполнению профессиональной деятельности. Задача ставится как классификация персонала на удовлетворяющих и не удовлетворяющих требованиям по результатам проверки компетенций персонала. Эксперт, проверяющий сотрудников, по каждому требованию каждой компетенции выставляет баллы по десятибалльной шкале. Также выставляется итоговая оценка готовности специалиста выполнять свои обязанности. Итоговая оценка годности сотрудника к выполнению профессиональной деятельности не определяется суммой баллов, минимальным баллом и т.п. Оценка учитывает важность компетенции, общее впечатление эксперта от ответов и другие факторы (может быть, даже впечатление эксперта). Оценка годности сотрудника представляет сложную трудно формализуемую задачу. В настоящее время известны различные аналитические решения для управления персоналом (People Analytics и Workforce Analytic) [4]. People Analytics и Workforce Analytic представляют собой глубокий анализ личных свойств работников для управления персоналом. В нашем случае задача представляет собой бинарную классификацию персонала на готовых и не готовых к выполнению профессиональной деятельности работников. Исходные данные являются числовыми значениями. Для решения таких задач эффективны полносвязные нейронные сети [5]. Поэтому было предложено для поддержки работы экспертов применить нейросетевую классификацию

Прежде чем строить нейронную сеть, необходимо сформировать обучающую выборку. Это множество обучающих примеров, содержащих признаки, описывающие отдельные объекты, и целевые переменные, метки. Для решаемой задачи признаки должны характеризовать степень готовности специалистов к выполнению своих функций, а целевое значение — итоговая оценка годности специалиста. Для оценки уровня готовности персонала авиационной безопасности к выполнению профессиональной деятельности в соответствии с требованиями Федеральной системы

подготовки персонала в области авиационной безопасности (АБ) [6] выбраны следующие признаки:

1. Правовые и нормативные акты по обеспечению АБ
2. Основные понятия, цель и основы обеспечения АБ
3. Состояние АБ в ГА. Административно-организационные аспекты.
4. Полномочия, роль и обязанности работников систем АБ. Организация работы систем АБ.
5. Система управления обеспечением АБ. Концепции обеспечения АБ
6. Обеспечение АБ при наземном обслуживании воздушных судов.
7. Организация пропускного и внутриобъектового режимов
8. Конфигурация аэропорта и охраняемые зоны ограниченного доступа
9. Меры АБ в пределах контролируемой и неконтролируемой зоны. Контроль доступа к воздушным судам.
10. Технические средства и оборудование
11. Предметы и вещества, запрещенные и (или) ограниченные к перемещению в контролируемую зону
12. Организация досмотра персонала, посетителей и транспортных средств
13. Правила проведения предполетного и послеполетного досмотров
14. Досмотр пассажиров с использованием технических средств. Контактный метод досмотра
15. Досмотр багажа и ручной клади с помощью рентгено-телевизионной системы безопасности (интроскопа)
16. Метод наблюдения и опроса (профайлинг)
17. Общие сведения о терроризме и актах незаконного вмешательства. Общие сведения о взрывных устройствах, взрывчатых веществах, оружие и боеприпасах. Обнаружение, опознавание, сокрытие.
18. Угрозы в деятельности гражданской авиации. Характеристики нарушителей. Психология поведения нарушителей.
19. Особенности кризисных ситуаций. Определение степени серьезности происшествия
20. Порядок действия системы АБ в нестандартных ситуациях
21. Программа противодействия незаконного вмешательства
22. Меры охраны и защиты воздушных судов на земле. Порядок проведения досмотра воздушных судов на земле
23. Контроль за выполнением технологических процессов обеспечения АБ
24. Управление кризисными ситуациями. Навыки решения конфликтных ситуаций
25. Методы проведения аудиторских и инспекционных проверок
26. Нормы, правила и процедуры обеспечения АБ

Анализ результатов проверок ведущих международных аэропортов Российской Федерации позволил получить 416 обучающих примера.

Следующим этапом формирования обучающей выборки является преобразование признаков, включающее:

- обработку аномальных значений, противоречий и дубликатов;
- восстановление пропущенных значений;
- преобразование данных.

В нашем случае исходными данными являются экспертные оценки результатов проверки специалистов. Набор проверок фиксированный. Экспертная оценка может лежать в диапазоне от 0 до 9, поэтому, в обучающей выборке отсутствуют аномальные значения, противоречия, дубликаты и пропущенные значения.

Преобразование (трансформация) данных заключается в кодировании данных и масштабировании — приведению к определенному диапазону. Нейронные сети работают только с числами, так как их работа основана на арифметических операциях взвешенной суммы. Поэтому все входные и выходные данные нейронной сети должны быть закодированы в виде чисел. Наш набор данных содержит признаки в виде чисел, которые не надо кодировать, и бинарные целевые значения: "годен" и "не годен". Закодируем значение "годен" единицей, а "не годен" — нулем. Используем масштабирование по минимаксу, или нормализацию [7]. В MATLAB нормализация выполняется автоматически.

Нейронная сеть предварительно исследовалась в системе MATLAB с использованием инструмента визуального программирования Neural Network Pattern Recognition App пакета Deep Learning Toolbox системы MATLAB [6].

Экспериментально была подобрана следующая структура сети: входной слой — 26 входных признаков, скрытый слой — 10 нейронов, выходной слой — 1 нейрон. В скрытом слое в качестве функции активации использован гиперболический тангенс, в выходном — функция softmax [7], т. е. выход сети представляет собой вероятность отнесения проверяемого сотрудника к классу "годен".

В MATLAB для задач классификации в качестве функции потерь используется перекрестная энтропия. Для обучения сети применяется масштабированный метод сопряженных градиентов (Scaled Conjugate Gradient). Для предотвращения переобучения использовалась перекрестная проверка с ранним остановом — валидация на отложенных данных [9]. При таком подходе исходное обучающее множество случайным образом разбивалось на три подмножества:

- обучающее множество (Training Set), которое используется для настройки весов и смещений сети (60% от общего числа обучающих примеров);
- тестовое множество (Test Set), которое используется для проверки качества обучения (10%);
- контрольное множество (Validation Set), которое используется в течение всего обучения, чтобы обнаружить явление переобучения (30%).

Обучение прекращается, когда функция потерь на контрольном множестве перестает уменьшаться.

Результаты обучения сети показаны на матрицах ошибок (рис. 1). В нашей задаче "истина" (выходной класс равен 1) — это сотрудники, удовлетворяющие требованиям. "Ложь" (выходной класс равен 0) — сотрудники, не удовлетворяющие требованиям. В матрицах ошибок показаны количество 4 исходов классификации: N_{TN} — количество истинно отрицательных исходов (фактический класс примера равен 0 и модель выдаст 0); N_{FP} — количество ложно положительных исходов (класс примера равен 0, а модель для него сформирует выход 1); N_{FN} — количество ложно отрицательных исходов (целевая переменная принимает значение 1, а на выходе модель выдаст 0); N_{TP} — количество истинно положительных исходов (фактический класс данного примера равен 1 и модель на выходе выдает 1).

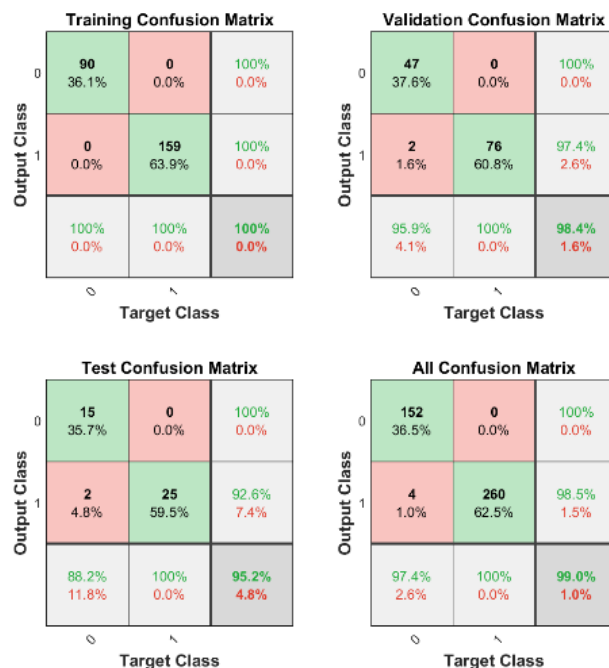


Рис. 1. Матрицы ошибок сети (Confusion matrix).

На матрицах показаны оценки точности (accuracy) — число правильно классифицированных наблюдений, отнесенное к общему числу наблюдений

$$R_{AC} = (N_{TP} + N_{TN}) / (N_{TP} + N_{TN} + N_{FP} + N_{FN}).$$

Матрицы ошибок показывают высокое качество классификации. На валидационном множестве сотрудники, не удовлетворяющие требованиям (класс 0), определены с точностью 100%. Сотрудники, удовлетворяющие требованиям (класс 1), определены с точностью 97,4%.

Рассчитаем основные показатели качества нейросетевого классификатора [10], используя матрицу ошибок (рис. 1). Расчет проведем по валидационному множеству. Из матрицы ошибок (рис. 1) получаем: $N_{TP} = 76$; $N_{FP} = 0$; $N_{TN} = 47$; $N_{FN} = 2$. Так как обучающая выборка не сбалансирована (количество примеров с разными исходами существенно отличается), то для оценки точности применим прецизионность (Precision), т. е. точность предсказания положительного исхода (сотрудников, удовлетворяющих требованиям): $R_{PR} = N_{TP} / (N_{TP} + N_{FP}) = 1,00$. Полнота (Recall), или чувствительность (Sensitivity), т. е. доля истинно положительных примеров равна 97,4% — из всех сотрудников, определенных как удовлетворяющие требованиям, 97,4% действительно удовлетворяют требованиям: $S_R = N_{TP} / (N_{TP} + N_{FN}) = 0,974$. Специфичность (Specificity), т. е. доля правильно классифицированных неудовлетворительных сотрудников составляет 95,9%: $S_P = N_{TN} / (N_{TN} + N_{FP}) = 0,959$. Таким образом, разработанный нейросетевой классификатор обеспечивает высокое качество классификации.

Пакет Deep Learning Toolbox системы MATLAB удобен для построения и исследования нейронной сети на множестве обучающих примеров, но не позволяет оценивать уровень готовности отдельного сотрудника. В MATLAB можно сохранить параметры обученной сети. Другая программа MATLAB, загрузив параметры сети и

получив данные сотрудника может оценить его готовность. Однако такой подход нецелесообразен по ряду причин: потребуются покупки дорогостоящих лицензий для каждого рабочего места; программа на MATLAB потребует специальных навыков пользователей по работе с системой MATLAB; программа будет предъявлять достаточно высокие требования к аппаратно-программной платформе.

Альтернативным подходом является разработка автономной программы на языке C++, что позволит: обеспечить низкую стоимость программы; получить исполняемый файл, который можно запускать на Windows-компьютерах различной конфигурации; использовать привычный для пользователя графический интерфейс, что минимизирует требования к пользователям. При разработке автономной программы целесообразно использовать подобранную в MATLAB структуру сети.

Разработанная программа Neurotest оценки готовности специалиста авиационной безопасности к выполнению профессиональных задач обеспечивает:

- ввод личных данных специалиста и ему выставленных баллов по каждому требованию компетенции;
- формирование итоговой оценки готовности специалиста;
- автоматическое сохранение информации результатах оценивания специалиста в базе данных;
- просмотр сохранённых результатов оценивания персонала.

После запуска программы Neurotest на экране появится главное окно приложения, которое представлено на рис. 2.



Рис. 2. Главное окно программы

Главное меню содержит четыре пункта: пункт "Тест" предназначен для ввода исходных данных и запуска процедуры оценивания; пункт "Результаты" предназначен для просмотра результатов выполненного и предыдущих оцениваний; пункт "Помощь" предназначен для вывода информации о программе; пункт "Выход" осуществляет выход из программы.

Для определения оценки готовности специалиста следует выбрать пункт меню "Тест", после чего появится диалоговое окно ввода личных данных специалиста (рис. 3).

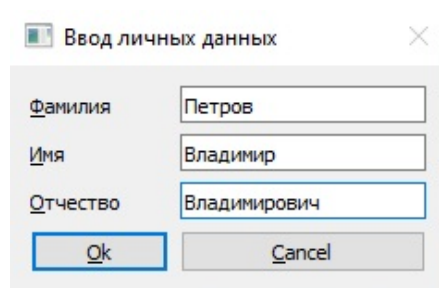


Рис. 3. Диалоговое окно для ввода личных данных

При вводе личных данных применяется валидация, согласно которой первая буква фамилии, имени или отчества должны быть прописной, остальные — строчными. Размер поля ввода равен 20 символам. Перемещение между полями ввода осуществляется с помощью клавиш курсора "↑" и "↓" и клавишей "Tab". Также выбрать поле для ввода или редактирования можно с помощью мыши. После заполнения всех полей ввода кнопка "Ok" становится активной, и после нажатия на неё текущее окно закроется и появится диалоговое окно для выставления баллов по требованиям компетенций. В окне для ввода баллов требуется занесение всех баллов и возможен ввод только цифр от 0 до 9.

После нажатия на кнопку "Ok" (она активна после занесения всех баллов) диалоговое окно закроется и введенный массив баллов будет использован для вычисления итоговой оценки готовности специалиста авиационной безопасности к выполнению профессиональных задач. Результат проверки специалиста в виде выставленной ему оценки "годен" или "отсев" выводится на экран с помощью информационного окна, которое представлено на рис. 4.

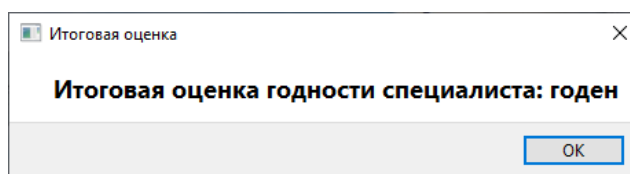


Рис. 4. Информационное окно

После вывода на экран оценки, информация о специалисте (фамилия, имя, отчество, выставленные баллы, итоговая оценка, а также дата и время оценивания) автоматически сохраняется в базе данных. При выборе пункта меню "Результаты" появится диалоговое окно (рис. 4), в котором отображаются результаты ранее проведенных проверок компетенций.

Представленная нейросеть решает уникальную задачу по классификации авиационного персонала. Специфика требований к персоналу приводит к заметной неопределенности и неоднозначности результатов тестирования в процессе аттестации, преодолеть которые весьма затруднительно, что заметно осложняет процесс принятия решения о готовности специалиста к производственной деятельности. За счет того, что в сети реализуется неалгоритмический анализ исходных данных, в процедуру решения удается включить неформальные, интуитивные представления эксперта по аттестации о

персонале за счет реализации специального обучения сети.

Предложенный вариант нейросети достаточно хорошо адаптируется к условиям использования в современных автоматизированных системах безопасности, основанных на информационном управлении.

Переход в системе управления авиационной безопасностью на компетентностную методологию оценки уровня профессиональной готовности персонала заметно повышает качество системы защиты объекта.

Список литературы

1. Sweet K. M. 2008 Aviation and Airport Security: Terrorism and Safety Concerns. — Boca Raton: CRC Press, 2008. — 384 p.
2. Price J. J. Practical Aviation Security: Predicting and Preventing Future Threats / J. Price, J. Forrest. — Oxford: Butterworth-Heinemann, 2016. — 598 p.
3. Овченков Н. И. Проблема человеческого фактора в области авиационной безопасности и некоторые подходы к ее решению / Н. И. Овченков, Д. В. Аверин // Научный вестник ГосНИИ ГА. — 2020. — № 30(341). — С. 107–116.
4. Khan N. Introduction to People Analytics: A Practical Guide to Data-driven HR / N. Khan, D. Milner, B. Marr. — London: Kogan Page, 2020. — 353 p.
5. Аггарвал Ч. Нейронные сети и глубокое обучение. — СПб.: ООО "Диалектика", 2020. — 752 с.
6. Федеральная система подготовки персонала в области авиационной безопасности (Национальная программа подготовки персонала в области авиационной безопасности), 2019. — 61 с.
7. Жерон О. Прикладное машинное обучение с помощью Scikit Learn и TensorFlow: концепции, инструменты и техники для создания интеллектуальных систем. — СПб.: "Альфа книга", 2018. — 688 с.
8. Beale M. H. Deep Learning Toolbox. User's Guide / M. H. Beale, M. T. Hagan, H. B. Demuth. — Natick: MathWorks, 2019. — 762 p.
9. Рашка С. Python и машинное обучение: машинное и глубокое обучение с использованием Python, Scikit-learn и TensorFlow. — СПб.: ООО "Диалектика", 2010. — 656 с.
10. Брюс П., Брюс Э. Практическая статистика для специалистов Data Science / П. Брюс, С. Брюс. — СПб.: БХВ-Петербург, 2018. — 304 с.

МЕТОД ОЦЕНКИ СПЛОЧЕННОСТИ ГИБРИДНОЙ ИНТЕЛЛЕКТУАЛЬНОЙ МНОГОАГЕНТНОЙ СИСТЕМЫ. ЧАСТЬ 2: СХОДСТВО ОНТОЛОГИЙ АГЕНТОВ¹

С.В.Листопад

Калининградский филиал Федерального исследовательского центра «Информатика и управление» Российской академии наук, ser-list-post@yandex.ru

Как показано в [1, 2], практические проблемы характеризуются слабой формализацией, сетевым характером условий и целей, субъективностью, динамичностью, неоднородностью, неполнотой и другими НЕ-факторами. Один из подходов к их решению без существенных упрощений – гибридные интеллектуальные многоагентные системы (ГиИМАС), которые благодаря наличию гибридной компоненты обеспечивают учет функциональной и инструментальной неоднородности, а за счет многоагентного подхода позволяют моделировать работу коллектива экспертов, с присущей ему разнородностью целей, субъективностью мнений и динамичностью отношений участников.

При этом кроме достоинств коллективного решения проблем ГиИМАС наследуют и отдельные его недостатки, например, сложность организации эффективного взаимодействия экспертов. Недостаточно подобрать экспертов в соответствии с их специальностями: существует риск возникновения деструктивных конфликтов, негативных групповых процессов и эффектов, например, конформизма, эффекта Рингельмана и других, в результате чего группа не в состоянии решить поставленную проблему в отведенные сроки [3]. Для стабильно качественного решения проблем за приемлемое время требуется сформировать из экспертов-индивидуалистов сплоченный коллектив с устоявшимися нормами взаимодействия, едиными целями и пониманием проблемы.

В случае с ГиИМАС в связи с высокой сложностью моделирования решаемых практических проблем требуется формировать систему из агентов, построенных различными независимыми коллективами разработчиков. Это обуславливает необходимость дополнительных трудозатрат на их интеграцию [4], так как агенты могут реализовывать различные протоколы решения проблемы, обмениваться сообщениями на разных языках, обладать противоречащими друг другу моделями предметной области. Для снижения этих трудозатрат предлагается реализовать механизмы моделирования процессов сплочения коллектива экспертов в рамках нового класса интеллектуальных систем – сплоченных гибридных интеллектуальных многоагентных систем (СГИМАС).

Групповые механизмы в коллективах, в том числе формирование сплоченности, изучаются в рамках групповой динамики [5]. Групповая сплоченность проявляется в создании единой социально-психологической общности людей, входящих в группу, и предполагает возникновение системы свойств группы, препятствующих нарушению ее психологической целостности [6]. Изучению процессов формирования сплоченного коллектива посвящены работы таких исследователей как А.И. Донцов [7], Р.Л. Кричевский [8], А.В. Сидоренков [9], Д. Картрайт [10], Г. Келли, Д. Тибо [11], Я. Морено [12], Ю.Л. Неймер [13], Т. Ньюком [14], Д. Тернер [15] и других. А.В. Петровским предложена стратометрическая концепция (СК) сплоченности группы [16], ставшая

¹ Работа выполнена при финансовой поддержке РФФИ (проект № 20-07-00104).

фундаментальной моделью, интегрирующей работы в этой области. Согласно ей, сплоченность группы рассматривается на трех уровнях (стратах):

- 1) страта эмоциональных межличностных отношений;
- 2) ценностно-ориентационное единство (ЦОЕ) – отношения опосредованы совместной деятельностью, на основе чего возникает единство основных ценностей;
- 3) ядро – члены группы разделяют цели групповой деятельности, поэтому здесь могут выявляться мотивы выбора членами группы друг друга, которые могут опосредоваться общими ценностями.

При моделировании данной концепции средствами ГиИМАС страта эмоциональных межличностных отношений не рассматривается, так как решение практических проблем, для которых создаются данные системы, не предполагает моделирование эмоций агентов. Таким образом, СГИМАС, моделирующая уровни ядра и ЦОЕ, описывается следующим выражением [4]:

$$chimas = \langle AG, env, INT, ORG, \{glng, ontng, protng\} \rangle, \quad (1)$$

где AG – множество агентов; env – концептуальная модель внешней среды СГИМАС; INT – элементы структурирования взаимодействий агентов, описываемые выражением (2); ORG – множество архитектур СГИМАС; $\{glng, ontng, protng\}$ – множество концептуальных моделей макроуровневых процессов: $glng$ – модель процесса согласования целей агентов между собой, обеспечивающая сплоченность на уровне ядра СК; $ontng$ – модель процесса согласования онтологий (моделей предметной области) агентов, соответствующая обмену знаниями, опытом и убеждениями между экспертами и моделированию страты ЦОЕ; $protng$ – модель процесса формирования протокола сплоченного взаимодействия, обеспечивающая согласование норм взаимодействия на уровне ядра СК.

Элементы структурирования взаимодействий из (1) описываются выражением

$$INT = \{prot_{bsc}, PRC, LANG, ont_{bsc}, chn\}, \quad (2)$$

где $prot_{bsc}$ – базовый протокол, обеспечивающий взаимодействие агентов по формированию протокола сплоченного взаимодействия для решения поставленных перед СГИМАС проблем; PRC – множество элементов для конструирования протокола решения проблем агентами-экспертами и агентом, принимающим решения; $LANG$ – множество языков передачи сообщений агентов; ont_{bsc} – базовая онтология, обеспечивающая интерпретацию агентами семантики сообщений по согласованию собственных моделей предметной области, целей, формированию протокола сплоченного взаимодействия, описываемая выражением (4); chn – степень сплоченности агентов (5).

Агент $ag \in AG$ из формулы (1) описывается выражением

$$ag = \langle id^{ag}, gl^{ag}, LANG^{ag}, ont^{ag}, ACT^{ag}, prt^{ag} \rangle, \quad (3)$$

где id^{ag} – идентификатор агента; gl^{ag} – нечеткая цель агента; $LANG^{ag} \subseteq LANG$ – множество языков, сообщения на которых могут быть записаны или прочитаны агентом; ont^{ag} – модель предметной области (онтология) агента, описываемая выражением (4); ACT^{ag} – множество действий, реализуемых агентом; prt^{ag} – модель протокола решения проблемы, разрабатываемого агентом.

Действие агента из множества ACT^{ag} описывается выражением

$$act^{ag} = \langle met_{act}^{ag}, it_{act}^{ag} \rangle,$$

где met_{ag} – метод решения проблемы; it_{ag} – интеллектуальная технология, в рамках которой реализован метод met_{ag} .

Таким образом, функция СГИМАС описывается выражением

$$act_{chimas} = \left(\bigcup_{ag \in AG^*} ACT_{ag} \right) \cup act_{col}, \quad \left| \bigcup_{ag \in AG} \bigcup_{act \in ACT^{ag}} it_{act}^{ag} \right| \geq 2,$$

где act_{col} – коллективная функция СГИМАС, конструируемая агентами динамически в соответствии с выработанным протоколом решения проблемы; ограничение требует, чтобы в СГИМАС использовалось не менее двух интеллектуальных технологий [17].

Модели базовой онтологий ont_{bsc} из выражения (2) и онтологий агентов ont^{ag} из выражения (3) описываются выражением

$$ont = \langle L, C, R, AT, FC, FR, FA, H^c, H^r, INST \rangle, \quad (4)$$

где $L = L^c \cup L^r \cup L^{at} \cup L^{va}$ – лексикон, множество лексем, состоящее из подмножеств лексем, обозначающих понятия L^c , отношения L^r , атрибуты L^{at} и их значения L^{va} ; C – множество концептов (понятий); $R: C \times C$ – множество отношений между концептами, первая компонента кортежа отношения называется доменом $dm(r) = Pp_1(r)$, а вторая – диапазоном значений отношения $rn(r) = Pp_2(r)$; $AT: C \times L^{va}$ – множество атрибутов концептов; $FC: 2^{L^c} \rightarrow 2^C$ – функция связи лексикона с концептами; $FR: 2^{L^r} \rightarrow 2^R$ – функция связи лексикона с отношениями; $FA: L^{at} \rightarrow AT$ – функция связи лексикона с атрибутами; $H^c = C \times C$ – таксономическая иерархия концептов; $H^r = R \times R$ – иерархия отношений; $INST$ – множество экземпляров, концептов единичного объема [18]. Функции FC и FR предполагают, что, в общем случае, одна лексема может соответствовать нескольким концептам или отношениям и, наоборот, один концепт или отношение может описываться несколькими лексемами.

Ключевая характеристика, описывающая состояние СГИМАС, – степень сплоченности ее агентов

$$chn = \langle gls, onts, protc \rangle, \quad gls, onts, protc \in [0, 1], \quad (5)$$

где gls – степень сходства целей агентов; $onts$ – степень сходства моделей предметной области агентов; $protc$ – степень согласованности протокола решения проблемы. Она используется как критерий оптимальности при согласовании целей act_{glnag}^{ag} и моделей предметной области act_{ontng}^{ag} , выработке протокола решения проблемы $act_{protnag}^{ag}$ агентами-экспертами и агентом, принимающим решения. Кроме того, она необходима для анализа агентом-фасилитатором текущей ситуации принятия решений в СГИМАС.

Сплоченность chn_{ij}^{ag} пары агентов $ag_i, ag_j \in (AG^{ex} \cup \{ag^{dm}\})$ рассчитывается путем вычисления соответствующих компонент кортежа, то есть gls_{ij}^{ag} , $onts_{ij}^{ag}$ и $protc_{ij}^{ag}$. Значения сплоченности каждой пары агентов СГИМАС формируют матрицу CHN^{ag} кортежей, определенных в соответствии с выражением (5). В результате, сплоченность системы в целом описывается выражением

$$chn_{chimas} = \sum_{i=1}^{nex+1} \sum_{j=1, j \neq i}^{nex+1} \frac{chn_{ij}^{ag}}{(nex+1)nex}. \quad (6)$$

Для оценки сходства целей агентов используется метод, предложенный в [4]. Настоящая работа посвящена другой части метода оценки сплоченности ГиИМАС, предназначенной для оценки сходства онтологий агентов.

Оценка степени сходства онтологий пары агентов $onts_{ij}^{ag}$ основывается на методе, предложенном в [19]. Согласно ему, сходство онтологий рассматривается на двух уровнях: вербальном и концептуальном. На вербальном уровне сравниваются лексиконы онтологий, на концептуальном – таксономии концептов и отношений.

Для агентов ag_i и ag_j , обладающих онтологиями ont_i^{ag} и ont_j^{ag} с лексиконами L_i и L_j , мера вербального сходства онтологий определяется выражением

$$LSO(ont_i^{ag}, ont_j^{ag}) = |L_i|^{-1} \sum_{l_k \in L_i} \max_{l_m \in L_j} LSL(l_k, l_m), \quad (7)$$

где LSL – сходство лексем, определяемое выражением

$$LSL(l_k, l_m) = \max(0, 1 - \frac{ed(l_k, l_m)}{\min(|l_k|, |l_m|)}),$$

в котором ed – редакторское расстояние Левенштейна [20], определяемое как количество символов, которое требуется добавить, удалить или изменить, чтобы получить из одной лексемы другую.

Для оценки сходства таксономий концептов вводятся понятия семантической котопии концепта, т.е. множества выше- и нижележащих концептов относительно него в таксономической иерархии H^c

$$SC^c(c_k, H^c) = \{c_m \in C \mid H^c(c_k, c_m) \vee H^c(c_m, c_k)\},$$

и семантической котопии множества концептов

$$SC^{cs}(C_p, H^c) = \bigcup_{u \in C_p} SC(u, H^c), \quad C_p \subseteq C.$$

Пересечение таксономий концептов онтологий ont_i^{ag} и ont_j^{ag} относительно лексемы $l \in L_i^c$ определяется в соответствии со следующими выражениями:

$$CTO(l, ont_i^{ag}, ont_j^{ag}) = \begin{cases} CTO'(l, ont_i^{ag}, ont_j^{ag}), & \text{если } l \in L_j^c, \\ CTO''(l, ont_i^{ag}, ont_j^{ag}), & \text{если } l \notin L_j^c, \end{cases}$$

$$CTO'(l, ont_i^{ag}, ont_j^{ag}) = \frac{|FC_i^{-1}(SC^{cs}(FC_i(l), H_i^c)) \cap FC_j^{-1}(SC^{cs}(FC_j(l), H_j^c))|}{|FC_i^{-1}(SC^{cs}(FC_i(l), H_i^c)) \cup FC_j^{-1}(SC^{cs}(FC_j(l), H_j^c))|},$$

$$CTO''(l, ont_i^{ag}, ont_j^{ag}) = \max_{u \in C_j} \frac{|FC_i^{-1}(SC^{cs}(FC_i(l), H_i^c)) \cap FC_j^{-1}(SC^{cs}(u, H_j^c))|}{|FC_i^{-1}(SC^{cs}(FC_i(l), H_i^c)) \cup FC_j^{-1}(SC^{cs}(u, H_j^c))|},$$

где $FC^{-1}: C \rightarrow L^c$ – функция, обратная FC , устанавливающая соответствие между концептом и описывающей его лексемой

В результате сходство таксономий концептов онтологий ont_i^{ag} и ont_j^{ag} определяется выражением

$$CTS(ont_i^{ag}, ont_j^{ag}) = |L_i^c|^{-1} \sum_{l_k \in L_i^c} CTO(l_k, ont_i^{ag}, ont_j^{ag}). \quad (8)$$

Оценка сходства таксономий отношений основывается на предположении, что схожие отношения имеют похожие домен и диапазон значений [19]. Для оценки таксономической близости концептов применяется мера на основе верхней котопии, т.е.

множества вершин, содержащего все вышележащие вершины (суперпонятия) по таксономической иерархии H^{con} по отношению к заданной вершине и саму вершину [21]

$$UC(c, H^c) = \{c_k \in C \mid H^c(c, c_k) \vee (c = c_k)\}.$$

Таксономическая мера близости понятий – отношение числа общих суперпонятий обеих вершин к числу всех суперпонятий обеих вершин

$$TS(c_k, c_m, H_k^c, H_m^c) = \frac{|FC^{-1}(UC(c_k, H_k^c)) \cap FC^{-1}(UC(c_m, H_m^c))|}{|FC^{-1}(UC(c_k, H_k^c)) \cup FC^{-1}(UC(c_m, H_m^c))|}. \quad (9)$$

Таким образом, сходство пары отношений определяется как геометрическое среднее меры близости (9) их доменов и диапазонов значений

$$S^R(r_k, r_m, H_k^c, H_m^c) = \sqrt{TS(dm(r_k), dm(r_m), H_k^c, H_m^c) \cdot TS(rn(r_k), rn(r_m), H_k^c, H_m^c)}.$$

Пересечение таксономий отношений онтологий ont_i^{ag} и ont_j^{ag} относительно лексемы $l \in L_i^r$ определяется следующими выражениями:

$$RTO(l, ont_i^{ag}, ont_j^{ag}) = \begin{cases} RTO'(l, ont_i^{ag}, ont_j^{ag}), & \text{если } l \in L_j^r, \\ RTO''(l, ont_i^{ag}, ont_j^{ag}), & \text{если } l \notin L_j^r, \end{cases}$$

$$RTO'(l, ont_i^{ag}, ont_j^{ag}) = |FR_i(\{l\})|^{-1} \sum_{r_i \in FR_i(\{l\})} \max_{r_j \in FR_j(\{l\})} S^R(r_i, r_j, H_i^c, H_j^c),$$

$$RTO''(l, ont_i^{ag}, ont_j^{ag}) = |FR_i(\{l\})|^{-1} \sum_{r_i \in FR_i(\{l\})} \max_{r_j \in R_j} S^R(r_i, r_j, H_i^c, H_j^c).$$

Сходство таксономий отношений онтологий определяется выражением

$$RTS(ont_i^{ag}, ont_j^{ag}) = |L_i^r|^{-1} \sum_{l_k \in L_i^r} RTO(l_k, ont_i^{ag}, ont_j^{ag}). \quad (10)$$

Таким образом, степень сходства онтологий пары агентов $onts_{ij}^{ag}$ может быть оценена как геометрическое среднее значений вербального сходства онтологий (7), сходства таксономий концептов (8) и отношений (10)

$$onts_{ij}^{ag} = \sqrt[3]{LSO(ont_i^{ag}, ont_j^{ag}) \cdot CTS(ont_i^{ag}, ont_j^{ag}) \cdot RTS(ont_i^{ag}, ont_j^{ag})}. \quad (11)$$

Как видно, выражения (7), (8), (10), на основе которых вычисляется степень сходства онтологий агентов (11), не обладают свойством симметричности. В результате, мера (11) также не обладает данным свойством, т.е. $onts_{ij}^{ag} \neq onts_{ji}^{ag}$, что учтено при вычислении сплоченности СГИМАС в целом по выражению (6). Таким образом, мера (11) применяется для оценки одной из компонент кортежа (5), используемого агентами-экспертами и агентом, принимающим решения, в качестве критерия эффективности их совместной работы, а также позволяющего агенту-фасилитатору вычислять показатель сплоченности системы в целом для идентификации ситуации решения проблемы и последующей выработки управляющих воздействий.

В работе представлена модель СГИМАС, в которой моделируется сплоченность агентов на двух из трех уровней стратометрической концепции А.В. Петровского путем согласования целей и моделей предметной области, а также выработки протокола решения проблемы агентами системы без внешнего управления этим процессом. Благодаря этому в интеллектуальной системе обеспечивается сплоченное поведение агентов, позволяющее преодолевать разногласия и избегать конфликтов, вызванных различиями в моделях проблемы и целях ее решения. Предложенная модель сплоченности повышает релевантность системы малым коллективам экспертов, решающим проблемы «за круглым столом», путем моделирования развития интеллектуальной системы и ее самоорганизации в сильном смысле [22]. Предложен

метод оценки сплоченности агентов, представлено детальное описание одной из его частей – оценка сходства онтологий агентов. Применение разработанной модели СГИМАС и метода оценки сплоченности агентов в ней позволит упростить разработку интеллектуальных систем за счет облегчения интеграции разнородных агентов, построенных независимыми разработчиками.

Список литературы

1. Нариньяни А.С. Инженерия знаний и НЕ-факторы: краткий обзор-08 / А.С. Нариньяни // Вопросы искусственного интеллекта. — 2008. — № 1. — С. 61–77.
2. Спиридонов В.Ф. Психология мышления: Решение задач и проблем: Учебное пособие / В.Ф. Спиридонов. — М.: Генезис, 2006. — 319 с.
3. Жуков Ю.М. Технологии командообразования: Учеб. пособие для студентов вузов / Ю.М. Жуков, А.В. Журавлев, Е.Н. Павлова. — М.: Аспект Пресс, 2008. — 320 с.
4. Листопад С.В. Метод оценки сплоченности гибридной интеллектуальной многоагентной системы. Часть 1: сходство целей агентов / С.В. Листопад // В: VIII Международная научно-практическая конференция «Нечеткие системы, мягкие вычисления и интеллектуальные технологии» НСМВИТ-2020 (29 июня – 1 июля 2020 г., г. Смоленск, Россия). Труды конференции. В 2-х томах. — Т. 2. – Смоленск: Универсум, 2020. – С. 16 – 25.
5. Lewin K. Field theory in social science: selected theoretical papers / K. Lewin. — New York: Harper, 1976. — 346 p.
6. Васильченко М.В. Психология личности и группы: Учеб. Пособие / М.В. Васильченко, М.П. Литвиненко, И.С. Мансурова и др. — Ростов-на-Дону: ДГТУ, 2011. — 174 с.
7. Донцов А.И. Проблемы групповой сплоченности / А.И. Донцов. — М.: МГУ, 1979. — 128 с.
8. Кричевский Р.Л. Проблема сплоченности малых групп в зарубежной социальной психологии / Р.Л. Кричевский // Вопросы психологии. — 1973. — № 3. — С. 174 – 184.
9. Сидоренков А.В. Микрогрупповая концепция / А.В. Сидоренков // Российский психологический журнал. — 2006. — № 1. — С. 21 – 40.
10. Cartwright D. The nature of group cohesiveness / D. Cartwright // In: Group dynamics: Research and theory. — New York: Harper & Row, 1968. — pp. 51-73.
11. Келли Г. Межличностные отношения. Теория взаимозависимости / Г. Келли, Д. Тибо // В: Современная зарубежная социальная психология. Тексты. Ред. Г.М. Андреева, Н.Н. Богомолова, Л.А. Петровская. — М.: МГУ, 1984. — С. 61 – 81.
12. Морено Я.Л. Социометрия: экспериментальный метод и наука об обществе / Я.Л. Морено. — М.: Акад. проект, 2001. — 384с.
13. Неймер Ю.Л. Сплоченность как характеристика первичного коллектива и ее социологическое измерение / Ю.Л. Неймер // Социологические исследования. — 1975. — №2. — С. 160 – 162.
14. Newcomb T. Varieties of interpersonal attraction / T. Newcomb // In: Group dynamics. — Evanston, IL: Row, Peterson, 1960. — pp. 104 – 119.
15. Тернер Д. Социальное влияние / Д. Тернер. — СПб.: Питер, 2003. — 256 с.
16. Петровский А.В. Опыт построения социально-психологической концепции групповой активности / А.В. Петровский // Вопросы психологии. — 1973. — №5. — С. 3 – 17.
17. Колесников А.В. Гибридные интеллектуальные системы с самоорганизацией: координация, согласованность, спор / А.В. Колесников, И.А. Кириков, С.В. Листопад. — М.: ИПИ РАН, 2014. — 189 с.
18. Крюков К.В. Меры семантической близости в онтологии / К.В. Крюков, Л.А. Панкова, В.А. Пронина и др. // Проблемы управления. — 2010. — № 5. — С. 2 – 14.
19. Maedche A. Measuring Similarity between Ontologies / A. Maedche, S. Staab // Lecture Notes in Artificial Intelligence. — 2002. — Vol. 2473. — pp.15 – 21.
20. Левенштейн В.И. Двоичные коды с исправлением выпадений, вставок и замещений символов // Доклады Академии наук СССР. — 1965. — Т. 163. — № 4. — С. 845 – 848.
21. Maedche A. Clustering Ontology-Based Metadata in the Semantic Web / A. Maedche, V. Zacharias // Lecture Notes in Artificial Intelligence. — 2002. — Vol. 2431. — pp. 348 – 360.
22. Serugendo G.D.M. Self-organization in multiagent systems / G.D.M. Serugendo, M.-P. Gleizes, A. Karageorgos // The Knowledge engineering review. — 2005. — Vol. 20(2) — pp. 165 – 189.

ПРОГНОЗИРОВАНИЕ ПОКАЗАТЕЛЕЙ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА ПРОИЗВОДСТВА АЛЮМИНИЯ НА ОСНОВЕ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Н.М. Луговая¹, А.С. Михалев¹, Т.Г. Пенькова²

¹ Сибирский федеральный университет, asmikhalev@yandex.ru

² Институт вычислительного моделирования СО РАН, penkova_t@icm.krasn.ru

На сегодняшний день алюминий занимает лидирующие позиции среди различных конструкционных материалов. Благодаря своим эксплуатационным качествам его использование растет во всех отраслях мировой экономики [1]. Производство алюминия является многоступенчатым процессом, включающим в себя производство сырья, получение алюминия электролизом криолитоглиноземного расплава и литье заготовок для дальнейшей переработки. В данной работе рассматривается производство алюминия в процессе электролиза.

Электролиз алюминия происходит в специальных ваннах – электролизерах. Для управления электролизёром используется блок управления, который измеряет такие параметры, как напряжение, значение текущего тока на каждом аноде, а также управляет двигателями привода анодов и пневматическими клапанами включения пробойников и дозаторов глинозема. Контроль таких параметров позволяет выявлять технологические нарушения в процессе эксплуатации электролизера.

Диагностика причин неэффективной работы электролизеров осуществляется постфактум, что является сдерживающим фактором для достижения наилучших технико-экономических показателей, как электролизной серии, так и всего предприятия. В связи с этим актуальной становится задача по разработке системы оперативного контроля состояния технологического процесса с применением современных научно-методологических подходов на основе прогнозной аналитики, которая позволит оценивать текущее состояние технологии, диагностировать причины снижения производительности электролизёров и прогнозировать нарушения по данным контролируемых параметров.

Описание системы надежности

Особые требования при разработке системы, функционирующей в режиме реального времени, когда в случае сбоя нет возможности восстановить исходное состояние и повторить исполнение задач, предъявляются к надежности программных средств и формируемых решений. При проектировании система контроля состояния технологического процесса разделена на три подсистемы, каждая из которых направлена на прогнозирование возникновения технологических нарушений по значениям контролируемых параметров. Первая подсистема осуществляет прогнозирование технологических нарушений на основе временного ряда напряжения. Вторая подсистема осуществляет прогнозирование на основе временного ряда токов 24 анодов электролизера. В задачи данной подсистемы также входит выявление «проблемного» анода, ставшего причиной возникновения нарушения. Третья подсистема осуществляет прогнозирование на основе временного ряда «шума». Все три подсистемы работают параллельно, получая на вход значения соответствующих временных рядов.

Для формирования решения на основе результатов работы подсистем применяются различные алгоритмы [2], наиболее распространенными из них являются алгоритмы голосования. Существует пять базовых алгоритмов голосования: голосование абсолютным большинством [3], два выхода из N -голосования [4], голосование согласованным большинством [5], нечеткое голосование согласованным большинством [6] и медианное голосование [7].

Во всех алгоритмах существует четкое различие между согласованностью и корректностью. Например, основная часть алгоритмов предполагает, что если большинство выходов «согласны», то основной вывод должен быть корректным [8]. Однако, это может привести к ошибочным результатам, особенно, в случае, когда количество возможных выходов очень мало. Например, предположим, что выходные данные каждой подсистемы представляют собой одну переменную, которая принимает значения 0 или 1. Это означает, что все ошибочные выходы автоматически согласуются [9], и очень вероятно, что если модули имеют ошибки, то может образоваться большинство согласованных ошибочных выходов. В общем случае рассмотрим несколько выходных переменных, каждая из которых принимает несколько значений. Общее количество разрешенных комбинаций выходных переменных и их значений определяет число p выходных состояний или мощность выходного пространства. Если имеется несколько правильных выходов, то простой избиратель бесполезен. Следовательно, при голосовании будем считать, что для каждого входа есть только один правильный результат. Из этого следует, что при заданной выходной мощности p мы имеем одно правильное состояние выхода [10], а $p - 1$ состояния ошибки. В данном контексте выбор среднего значения (или простое медианное голосование) представляет собой простую альтернативу вынесения решения, где медианой всех выходных значений выбран правильный ответ [11].

Учитывая все особенности существующих алгоритмов, на данном этапе разработки системы принято решение выбрать модификацию медианного алгоритма голосования [12]. Модификация заключается во введении динамической оценки надежности или «веса», имеющей элемент забывания. Вес представляет собой среднее результатов голосования. Обработка происходит следующим образом: для каждой подсистемы создается булевый стек, заданной длины, в который добавляется 0 – в случае, если блок голосования решает, что получен неверный ответ; 1 – в случае, если ответ верный или происходит нечеткое голосование, когда принадлежность значения выхода к выигравшему классу больше нуля. В качестве оценки надежности может рассматриваться точность моделей прогнозирования значений технологических параметров, являющихся «ядром» каждой из подсистем. Учитывая переобучение моделей, данная оценка будет оставаться актуальной.

Описание исходных данных и предобработка

Исходные данные представляют собой временные ряды параметров, фиксируемых датчиками на опытном участке Саяногорского алюминиевого завода: напряжение, ток на 24 анодах, «шум». Под «шумом» понимается величина амплитуды колебаний приведенного напряжения электролизера за определенный период. При нормальной работе электролизера колебания напряжения обусловлены сходом газовых пузырей с «подошвы» анода и имеют незначительную амплитуду. Изменение характера и величины значений параметров является индикатором возникновения технологических нарушений.

Временные ряды напряжения содержат трехминутные значения за период с 01 марта 2020 г. по 31 марта 2020 г. Временные ряды тока на анодах содержат десятисекундные значения за период с 01 марта 2020 г. по 31 марта 2020 г. Временные ряды «шума» содержат пятиминутные значения за период с 01 марта 2020 г. по 31 марта 2020 г. Общий объем выборки по напряжению составляет 120 тыс. записей, по токам – более 2 млн. записей, по «шумам» – 65 тыс. записей. Для настройки моделей прогнозирования и оценки ее качества выборки данных были разделены на три части: обучающую, валидационную и тестовую.

Этапу построения прогнозной модели предшествует этап анализа и предварительной обработки временного ряда. Предварительная обработка временных рядов заключается в выявлении аномальных значений и сглаживании ряда. Причинами появления аномальных значений могут быть технические ошибки, возникающие при сборе, обработке и передаче данных. Очистка данных от аномальных значений представляет специальный механизм для выявления и удаления явных несоответствий и других возможных ошибок в данных. Для поиска аномальных значений в работе использован алгоритм изолирующего леса [13].

Подбор архитектуры нейронных сетей

Для построения прогнозных моделей выбраны рекуррентные нейронные сети [14]. В рекуррентной нейронной сети элементы образуют направленную последовательность, благодаря чему появляется возможность обрабатывать серии событий во времени или последовательные пространственные цепочки. В отличие от многослойных перцептронов, рекуррентные сети могут использовать свою внутреннюю память для обработки последовательностей произвольной длины.

Эффективность решения задачи прогнозирования временных рядов с использованием искусственных нейронных сетей определяется их гиперпараметрами. Основными гиперпараметрами искусственной нейронной сети являются количество слоев и количество нейронов на каждом слое.

Для получения долгосрочного прогноза в работе используется прямой подход (Рис. 1). При прямом подходе для получения прогноза на $t + l$ шагов вперед применяются l обученных моделей. Каждая модель обучается для своего шага по временному ряду.

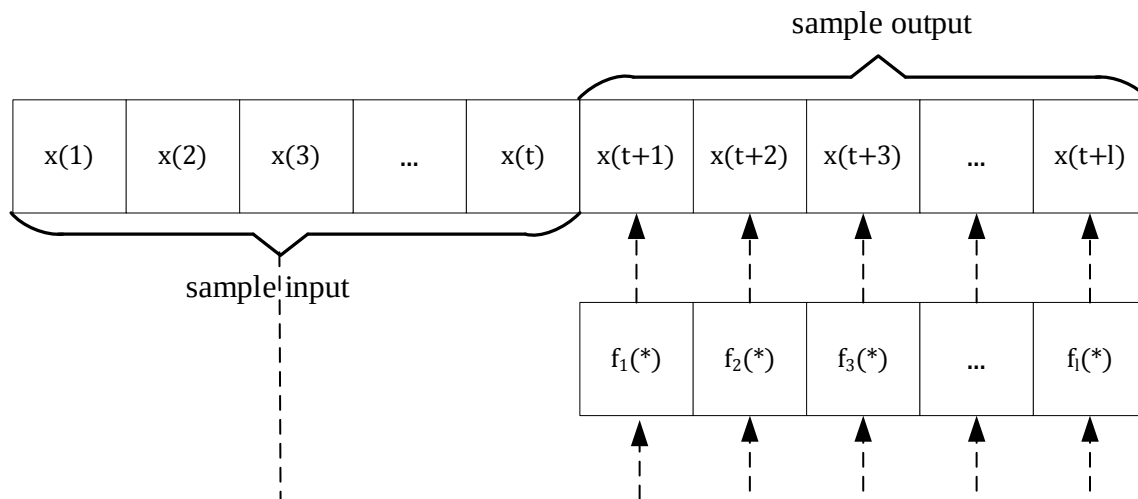


Рис. 1. Схема прямого подхода прогнозирования

Предсказания получаются путем применения l обученных моделей $\hat{f}_\tau, \tau = \overline{1, l}$:

$$\hat{x}(t + \tau) = \hat{f}_\tau(x(t - k + 1), x(t - k + 2), \dots, x(t))$$

В табл. 1-3 представлены архитектуры нейронных сетей для каждой подсистемы. Настройка других гиперпараметров модели осуществлялась с помощью техники случайного поиска с перекрестной проверкой.

Таблица 1

Характеристика архитектур нейронной сети. Временной ряд напряжения

Архитектура / Шаг	1	2	3	4	5	6	7	8	9	10
Количество слоев	3	3	2	3	3	2	3	3	3	3
Количество нейронов	60	80	100	70	80	90	80	100	90	90

Таблица 2

Характеристика архитектур нейронной сети. Временной ряд тока на анодах

Архитектура / Шаг	1	2	3	4	5	6	7	8	9	10
Количество слоев	3	3	3	3	3	3	3	3	3	3
Количество нейронов	80	80	70	80	80	70	90	80	90	90

Таблица 3

Характеристика архитектур нейронной сети. Временной ряд для «шума»

Архитектура / Шаг	1	2	3	4	5	6	7	8	9	10
Количество слоев	3	3	3	3	3	3	3	3	3	3
Количество нейронов	60	60	50	60	70	80	70	60	70	70

Прогнозирование

Входная последовательность для получения прогноза была выбрана длиной 10. Прогноз выполнялся на 10 тактов вперед. Результаты прогнозирования на тестовой выборке приведены на рисунках 2, 3 и 4.

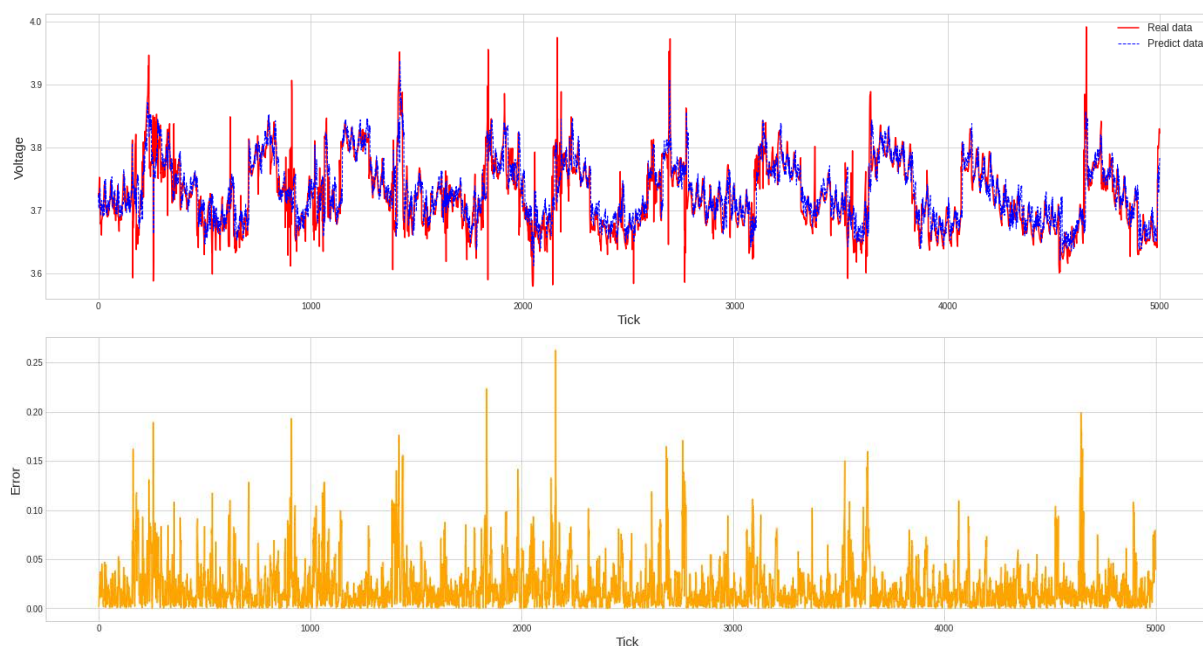


Рис 2. Результаты прогнозирования значений напряжения

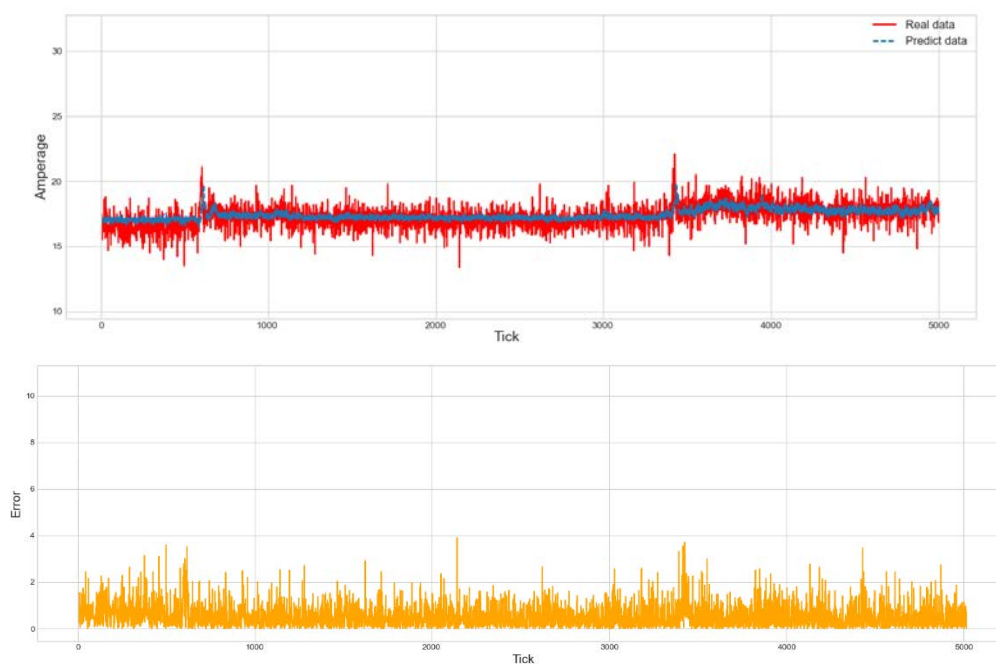


Рис 3. Результаты прогнозирования значений тока на анодах

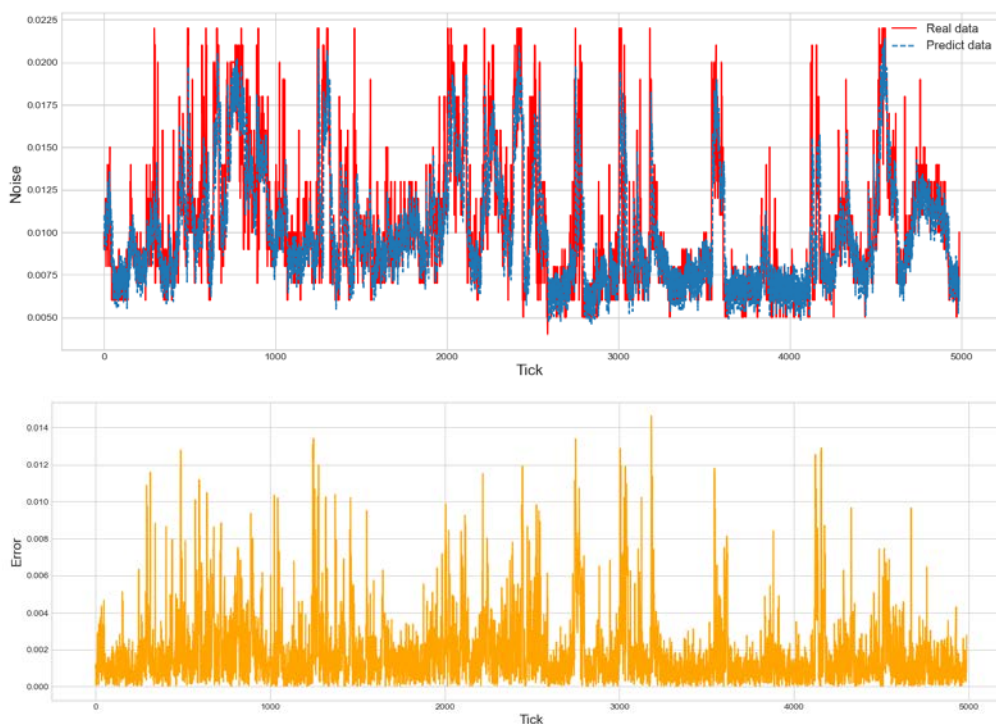


Рис 4. Результаты прогнозирования значений «шума»

Из анализа полученных графиков можно сделать вывод, что прогнозные модели позволяют определять тенденцию изменения контролируемого параметра с хорошей точностью и заблаговременно идентифицировать нарушения в технологическом процессе.

Заключение

В работе представлены теоретические аспекты построения системы поддержки принятия решений для оценки технического состояния и эффективности процесса

производства алюминия. Приведены результаты применения искусственных нейронных сетей для прогнозирования значений параметров технологического процесса производства алюминия: напряжения, силы тока на анодах и «шума». Для построения прогнозных моделей выбраны рекуррентные нейронные сети. Методом максимальной точности подобрана архитектура нейронной сети для каждого показателя. Результаты, полученные в процессе тестирования, являются допустимыми для предсказания значений показателей технологического процесса. Выявленные отклонения позволят своевременно принять меры по предотвращению возникновения технологических нарушений и, как следствие, повысить эффективность производства алюминия.

Список литературы

1. Дампилон Ж.В. Влияние производства алюминия в России на окружающую среду // Вестник ЧГУ. – 2008. – №3.
2. Song X., Zhai Z., Lv R., Guo Y., Zhu P. Optimal Design of the k-Out-of-n: G(F) Majority Voter // IEEE Access. – 2017. – Vol. 5. – P. 22647-22656.
3. Lakshmi P., Kirubanandasarathy N. A novel fault tolerant voter for identical core in satellite application // International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), 2016 – P. 814–818.
4. Avizienis A. The N-Version Approach to Fault -Tolerant Software // IEEE Transactions on Software Engineering. – 1985. – Vol. SE-11. – No. 12. – P. 1491-1501.
5. Jiao P., Xu K., Sun L. Diversity voting and its application in real-time strategy games multi-objective optimization decision-making behavior modeling // 3rd International Conference on Systems and Informatics (ICSAI), 2016. – P. 552–559
6. Kim K., Vouk M.A. McAllister D.F. Fault-Tolerant Software Voters Based on Fuzzy Equivalence Relations / North Carolina State University, 1997.
7. Лебедев В.В., Чернышев О.Л. Тестирование надёжности программного обеспечения системы автоматического управления // Информационные ресурсы и системы в экономике, науке и образовании: сборник статей VII Международной научно-практической конференции, 2017. – С. 55-60.
8. Slačka J., Halás M. Safety critical RTOS for space satellites // 20th International Conference on Process Control (PC), 2015. – С. 250- 254.
9. Nassaj A., Barabady J. Fault tree analysis of oil and gas distillation tower and application of Bayesian Networks // IEEE International Conference on Industrial Engineering and Engineering Management. – 2016. – Vol. 2016-December, 7798161. – С. 1669-1673.
10. Broen R.B. New Voters for Redundant Systems, Transactions of the ASME // Journal of Dynamic Systems, Measurement, and Control. – 1975. – No 3. – P. 41-45.
11. Сарамуд М.В., Зеленков П.В., Брезницкая В.В., Ковалев И.В., Ковалев Д.И. К вопросу надёжности программного обеспечения отказоустойчивых систем // Материалы Международной научно-практической конференции «Современное состояние науки и техники», 2016. – С. 149-151.
12. Kovalev I.V., Zelenkov P.V., Losev V.V., Kovalev D.I., Ivleva N.V., Saramud M.V. Multiversion environment creation for control algorithm execution by autonomous unmanned objects // IOP Conference Series: Materials Science and Engineering. 5. Series: "V International Workshop on Mathematical Models and their Applications 2016". – 2017. – P. 012025.
13. Liu F.T., Ting K.M., Zhou Z.-H. Isolation forest // ICDM 08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, 2008. – P.413-422.
14. Goodfellow I., Bengio Y., Courville A. Deep learning. – Cambridge: MIT press, 2016. 787 p.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ НЕЙРОСЕТЕВЫХ КЛАССИФИКАТОРОВ ДЛЯ РАСПОЗНАВАНИЯ ФАЗ ПОХОДКИ ЧЕЛОВЕКА

М.Е. Марков¹, В.В. Варлашин^{1,2}, А.В. Бахшиев², В.Н. Уланов¹

¹Санкт-Петербургский политехнический университет Петра Великого, Институт машиностроения, материалов и транспорта, maxxmarkov@gmail.com
²Центральный научно-исследовательский институт робототехники и технической кибернетики (ЦНИИ РТК), v.varlashin@rtc.ru

Мягкие экзоскелеты являются относительно новым направлением в развитии робототехники. Они применяются для снижения метаболической стоимости в процессе выполнения физической работы путем оказания помощи оператору. В частности, рассматривается задача помощи оператору в процессе ходьбы. Типовые варианты конструкций таких экзоскелетов представлены в работах [1-3]. Они включают в себя мягкий тканевый каркас, который обеспечивает фиксацию приводов, датчиков и тросовой системы на теле оператора. В отличие от экзоскелетов с жестким каркасом они обеспечивают большую подвижность в процессе ходьбы, поэтому основной задачей системы управления мягкого экзоскелета является распознавание фаз походки человека [4] для создания вспомогательных усилий в нужные моменты времени.

Согласно [4], фазы походки человека однозначно связаны с углами поворота суставов ног и их угловыми скоростями. Зависимости углов и угловых скоростей имеют характерные особенности для каждой фазы внутри цикла шага, что делает возможным применение углов и скоростей в качестве входных данных для классификатора походки. На рисунке 1 изображена схема расположения шестиосевых инерциальных датчиков на теле оператора. Каждый датчик включает в себя трехосевые гироскоп и акселерометр и позволяет считывать показания на частоте 1 кГц. Для преобразования измеренных угловых скоростей и ускорений в углы поворота суставов используется фильтра Калмана.

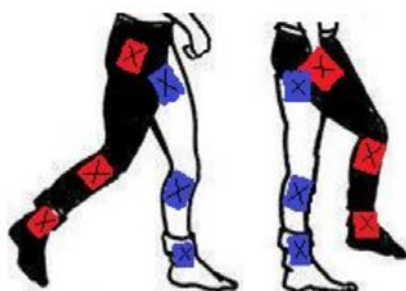


Рис.1. Схема расположения инерционных датчиков на теле оператора

Для формирования обучающих выборок для искусственных нейронных сетей (ИНС) реализовано программное обеспечение в пакете Matlab, решающее две задачи: нахождение отдельных шагов в экспериментальных данных и размечающее их в соответствии с фазами походки, представленными в работе [4]. Разбиение на фазы осуществляется в автоматическом режиме путем применения теоретических данных о средней длительности каждой фазы шага внутри цикла. На рисунке 2 представлены зависимости углов поворота суставов оператора от времени цикла шага для скорости движения 4 км/ч. Захват углов поворота осуществлялся в процессе ходьбы на беговой дорожке с заданной скоростью движения полотна (от 2 до 7 км/ч). Черным цветом на

рисунке 2 выделены средние зависимости углов поворота по всем шагам. Наибольшим разбросом обладает угол поворота в голеностопном суставе вследствие ударных воздействий и больших изменений в показаниях акселерометров в момент контакта с опорной поверхностью. Результаты для левой и правой ноги имеют небольшие отличия ввиду естественных физиологических особенностей испытуемого.

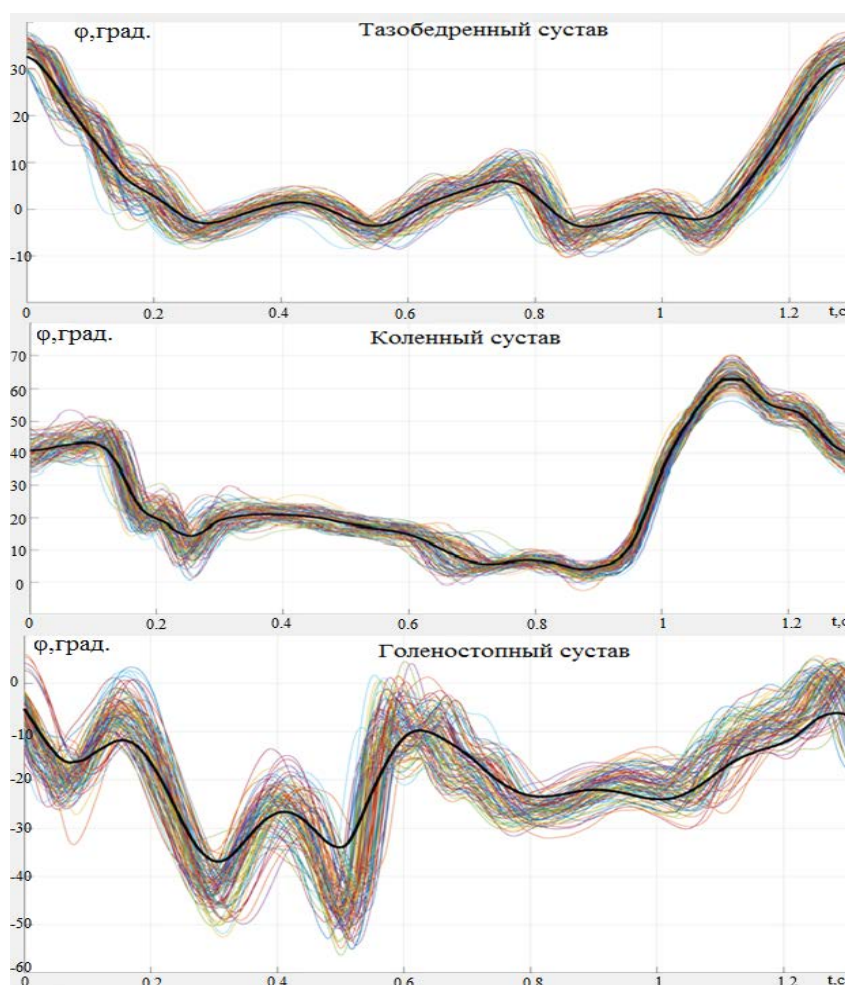


Рис.2. Экспериментальные зависимости углов поворота суставов от времени цикла шага для скорости

Определение фазы походки человека является ключевой задачей на этапе разработки мягкого экзоскелета и требует применения современных методов, в частности, использование нейросетевых классификаторов. Работа посвящена сравнению трех различных архитектур:

- ИНС прямого распространения;
- рекуррентная ИНС;
- каскадная ИНС.

Рекуррентная и каскадная ИНС представлены в двух вариантах: с количеством нейронов, аналогичным ИНС прямого распространения, и с количеством параметров, аналогичным ИНС прямого распространения. Разработка и обучение ИНС осуществляются в пакете Deep Learning Toolbox (ранее Neural Network Toolbox).

Каждый скрытый нейрон ИНС представлен гладкой нелинейной функцией активации (гиперболический тангенс). Выбор такой формы функции обусловлен тем, что в ней учитывается восстановительная фаза биологического нейрона. Для обучения ИНС выбран алгоритм обучения с учителем и алгоритм обратного распространения ошибки.

Для получения хорошего обобщения размер обучающей выборки, количество свободных параметров нейронной сети и допустимая ошибка классификации связаны эмпирическим правилом Видроу (1) [5, с.112].

$$N = O\left(\frac{W}{\varepsilon}\right) \quad (1)$$

где $O(\cdot)$ – порядок заключаемой величины,

W – общее количество свободных параметров ИНС,

ε – допустимая ошибка классификации.

Для определения ошибок аппроксимации вводится понятие эмпирического риска R [26, с.932], а верхняя граница риска определяется неравенством (2) [27, с.70].

$$R \leq O\left(\frac{C_f^2}{m_1}\right) + O\left(\frac{m_0 m_1}{N} \log N\right) \quad (2)$$

где m_0 – количество входных узлов,

m_1 – количество скрытых нейронов,

C_f – первый абсолютный момент изображения Фурье целевой функции,

N – размер обучающего множества.

В формуле (2) первое слагаемое точность наилучшей аппроксимации, а второе – точность эмпирического соответствия аппроксимации. Изменение m_1 оказывает противоположное влияние на слагаемые в правой части неравенства. Исходя из формулы (2), пренебрегая логарифмическим множителем, размер обучающего множества можно оценить по формуле (3)

$$N = O\left(\frac{m_0 m_1}{\varepsilon_0}\right) \quad (3)$$

где ε_0 – среднеквадратичное значение ошибки оценивания.

С учетом проведенных экспериментов по получению экспериментальных зависимостей можно определить необходимое количество свободных параметров ИНС по формуле (4)

$$W = C_f \sqrt{\frac{N}{m_0 \log N}} \quad (4)$$

Известно, что для решения сложных задач количество скрытых слоев должно быть не менее двух. Первый слой ИНС извлекает локальные признаки, последний извлекает глобальные признаки, обобщая выходные сигналы предыдущих слоев.

Количество входных узлов m_0 в данной работе равно семи. Входными данными выступают три угла поворота в суставах и три угловых скорости, а также евклидова норма показаний акселерометров датчика, закрепленного вблизи голеностопного сустава. Количество шагов для каждой скорости движения равно 100, а частота дискретизации при получении данных с акселерометров составляет 1 кГц. Тогда размер обучающего множества рассчитывается по формуле (5)

$$N = \sum_{i=1}^n \nu \cdot t_i \quad (5)$$

где n – количество шагов, используемых для обучения,

ν – частота дискретизации установки,

t – время выполнения шага.

Оценка количества свободных параметров ИНС для обучения рассчитывается по формуле (4), полученные значения приведены в таблице 1.

Таблица 1

Оценка количества настраиваемых параметров ИНС

Скорость движения, км/ч	Среднее время шага, с	Оценка количества параметров W
2	2.07	1566
3	1.65	1268
4	1.31	1358
5	1.13	1219
6	1.05	1423
7	0.89	1486

При увеличении скорости движения количество параметров ИНС, требующееся для классификации, снижается из-за уменьшения размера обучающего множества, а затем начинает возрастать из-за уменьшения гладкости целевой функции при больших скоростях. Для выявления ИНС, лучше решающей задачу классификации, число параметров выбрано заведомо большее. Для ИНС прямого распространения число свободных параметров определяется соотношением (6)

$$W = \sum_{i=1}^K (1 + m_{i-1}) m_i \quad (6)$$

где K – количество слоев сети,

m_i – количество нейронов на i -ом слое.

Приняв количество параметров ИНС равным 2500, а число слоев равным 4, получим следующее распределение нейронов по слоям: на первом слое 16 нейронов, на втором слое 24, на третьем слое 40, а на четвертом слое 20 нейронов, тогда $W = 2442$. Архитектуры рекуррентных ИНС и каскадных ИНС, сравниваемые с ИНС прямого распространения, имеют либо аналогичное распределение нейронов по слоям, либо аналогичное число параметров. Для сравнения нейронных сетей по критерию точности

классификации использована обучающая выборка для скорости 4 км/ч. Результаты обучения представлены в таблице 2.

Таблица 2

Оценка количества настраиваемых параметров ИНС

ИНС и ее распределение нейронов (параметров)	Минимальная точность классификации	Максимальная точность классификации	Средняя точность классификации
Прямого распространения	70.71	97.57	92.91
Рекуррентная*	71.01	98.29	94.2
Каскадная*	72.35	98.17	93.83
Рекуррентная**	71.27	97.27	92.88
Каскадная**	70.45	98.03	91.99

* с распределением нейронов, аналогичным ИНС прямого распространения,

** с количеством параметров, аналогичным ИНС прямого распространения.

Рекуррентная ИНС имеет наилучшие показатели по точности классификации, это связано с тем, что ИНС запоминает и учитывает состояния на предыдущих шагах обучения. При уменьшении количества настраиваемых параметров точность классификации снижается, однако остаётся выше всех обученных ИНС с тем же количеством настраиваемых параметров. Точность PrecisionW и полнота RecallW для каждого класса обученной рекуррентной нейронной сети на скорости 4 км/ч представлена в таблице 3.

Таблица 3

Точность и полнота для каждого класса обученной рекуррентной сети

Номер класса	Минимальная точность классификации	Максимальная точность классификации
1	0,75	0,65
2	0,79	0,78
3	0,88	0,94
4	0,97	0,93
5	0,35	0,28
6	0,95	0,96
7	0,98	0,95
8	0,94	0,92

На рисунке 3 представлена зависимость выходной функции ошибки определения фазы шага от цикла шага. По среднему значению функции ошибки можно судить о том, на каких фазах нейронная сеть запаздывает в оценке фазы шага или опережает целевую функцию при переходе на следующую фаз. Отрицательные числа соответствуют запаздыванию, а положительные - опережению.

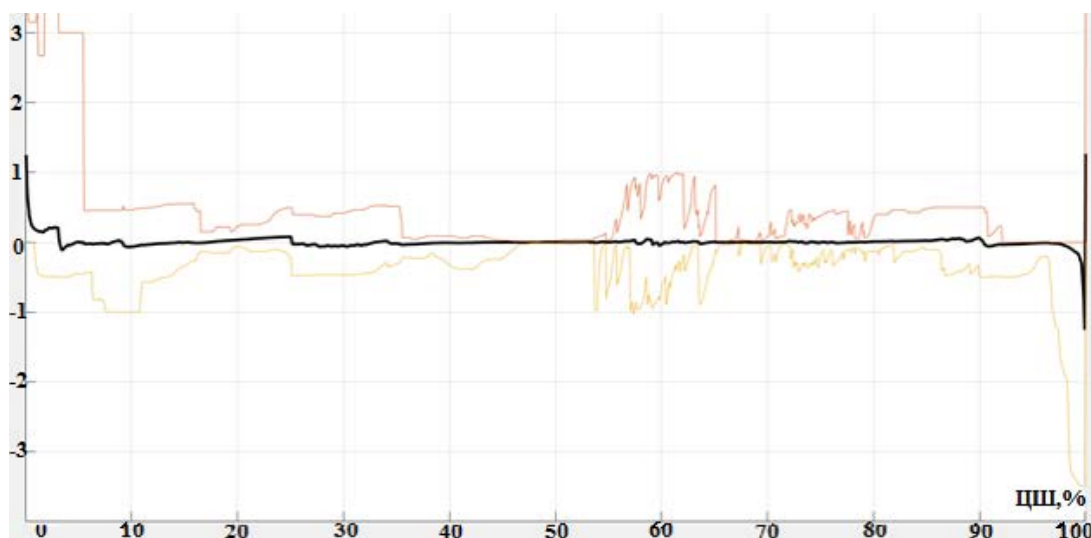


Рис.3. Зависимость выхода функции ошибки от цикла шага.

Анализируя зависимость можно сделать вывод о плавном переходе выхода нейронной сети между первой и второй, второй и третьей, пятой и шестой, шестой и седьмой фазами шага. Функция ошибки при этом плавно возрастает до 0.5, после границы фазы терпит скачок до -0.5, после чего плавно возрастает до нуля. Для перехода с третьей на четвертую и с четвертой на пятую фазу наблюдается запаздывание выхода нейронной сети относительно целевой функции. Для перехода с седьмой на восьмую фазу наблюдается опережение выхода нейронной сети относительно целевой функции.

Полученные в работе данные позволяют сделать вывод о том, что применение нейросетевых классификаторов фазы ходьбы человека в составе контура управления мягкого экзоскелета позволяет правильно определять моменты включения приводов для оказания помощи оператору. Полученные результаты являются частным случаем обучения ИНС, однако увеличение экспериментальной выборки путем использования данных, полученных на других скоростях движения оператора, позволят обобщить результаты.

Исследование показало, что наиболее эффективным вариантом архитектуры является рекуррентная ИНС, которая при аналогичном количестве независимых параметрах позволяет получить результаты лучше, чем в ИНС прямого распространения и каскадных ИНС.

Список литературы

1. Quinlivan B.T. Assistance magnitude versus metabolic cost reductions for a tethered multiarticular soft exosuit // Science Robotics. 2017 – Vol. 2 – № 2.
2. Asbeck A.T. Multi-joint soft exosuit for gait assistance // Proc. – IEEE International Conference on Robotics and Automation. – 2015 – pp. 6197–6204.
3. Ding Y. IMU-based iterative control for hip extension assistance with a soft exosuit // Proc. - IEEE International Conference on Robotics and Automation. – 2016. – pp. 3501–3508.
4. Скворцов Д.В.В. Клинический анализ движений. Анализ походки. // НПЦ «Стимул», Иваново. – 1996. – 344 с
5. Widrow B., Stearns S. Adaptive Signal Processing. Englewood Cliffs, NJ: Prentice-Hall, 1985. – 474 p.

ЧИСЛЕННОЕ ИССЛЕДОВАНИЕ ОСНОВАННОГО НА MCTS МЕТОДА MOTiFS ДЛЯ ОТБОРА ПРИЗНАКОВ

Б.В.Олейников¹, Н.И.Кожевников²

¹Сибирский федеральный университет,

Институт математики и фундаментальной информатики, oleynik48@mail.ru

² Сибирский федеральный университет,

Институт математики и фундаментальной информатики, nkozhevnikov@outlook.com

Методы понижения размерности исходных данных являются одним из основных моментов анализа данных, особенно в областях исследований, связанных с Big Data и Machine Learning, где описания наборов данных могут включать в себя от нескольких единиц до десятков миллионов доступных переменных (см., например, набор данных kdd2012 [1]). В настоящее время в задачах снижения размерности существуют 2 основных направления: задача выделения признаков (feature extraction) и задача отбора признаков (feature selection) [2].

Задача выделения признаков заключается в поиске нового множества признаков меньшего размера, чем изначальное множество, при этом новые признаки являются некоторыми функциями от исходных признаков рассматриваемого набора данных. Основные проблемы этого подхода - интерпретируемость полученных признаков, а также трудности «перевариваемости» больших наборов данных, когда размер исходного набора данных превышает объем памяти вычислительной системы.

Задача же отбора признаков большей частью менее связана с указанными проблемами и состоит в том, чтобы выбрать из существующего множества признаков некоторое их подмножество.

И в том и другом направлении разработано достаточно много методов, с тем или иным качеством решающих основную задачу понижения размерности [3]. В частности, в первом направлении используются линейный дискриминантный анализ, многомерное шкалирование, метод главных компонент, факторный анализ и другие «классические» методы или их модификации. Методов второго направления по понятным причинам значительно больше и, обычно, они представлены пятью классами, отличающимися тем, как они взаимодействуют с выбранным методом классификации или регрессии (основные задачи machine learning):

- методы фильтрации (filter methods) – методы, которые «фильтруют» избыточные или несущественные признаки, возвращая только те признаки, которые следует применять при классификации или прогнозировании;
- методы обертки (wrapper methods) – методы, направленные на то, что отбор признаков «обертывается» процедурой их поиска над методом классификации или регрессии;
- встраиваемые методы (embedded methods) – эти методы не разделяют в себе отбор признаков и классификацию/прогнозирование, а производят отбор прямо в процессе решения этих задач;
- гибридные методы (hybrid methods) - представляет собой гибрид из фильтров и оберток или встраиваний, объединённых в двухфазный процесс: сначала признаки фильтруются по статистическим свойствам, а затем применяются методы обертки или встраивания.
- ансамблевые методы (ensemble methods) - применяются больше для наборов данных с очень большим числом признаков. В данном подходе для начального множества признаков создается несколько подмножеств признаков, и эти группы каким-то образом объединяются, чтобы получить набор наиболее релевантных

признаков. Для этой группы методов можно применять различные способы выбора признаков и объединения их подмножеств, что придает им определенную гибкость.

Несмотря на то, что методов второго направления разработано достаточно много, в настоящее время интенсивно ведутся работы по созданию и исследованию новых. Одним из перспективных направлений является использование деревьев решений, построенных с помощью алгоритма Монте Карло деревьев случайного поиска - Monte-Carlo Tree Search (MCTS).

Настоящая работа направлена на исследование относительно недавно (2018г.) предложенного алгоритма отбора признаков Monte-Carlo Tree Search in Feature Selection (MOTiFS) [4], основанного на использовании MCTS, успешное применение которых в играх началось сравнительно недавно (2006 г.) и достигло впечатляющих результатов (игра Го [5]). MOTiFS является методом обертки над используемым методом классификации или регрессии.

Актуальность данной работы обусловлена отсутствием каких-либо сведений в открытом доступе по программной реализации алгоритма MOTiFS и исследованию эффективности его использования.

Отбор признаков в MOTiFS представляется как игра с одним игроком. Основные изменения в нем по сравнению с MCTS коснулись этапов выбора и расширения узлов в дереве решений, при этом каждый узел на глубине i в дереве определяет, выбран ли признак f_i . В дальнейшем построение асимметричного дерева решения осуществляется на основе алгоритма MCTS, который состоит из следующих 4 шагов (см. рис.1).

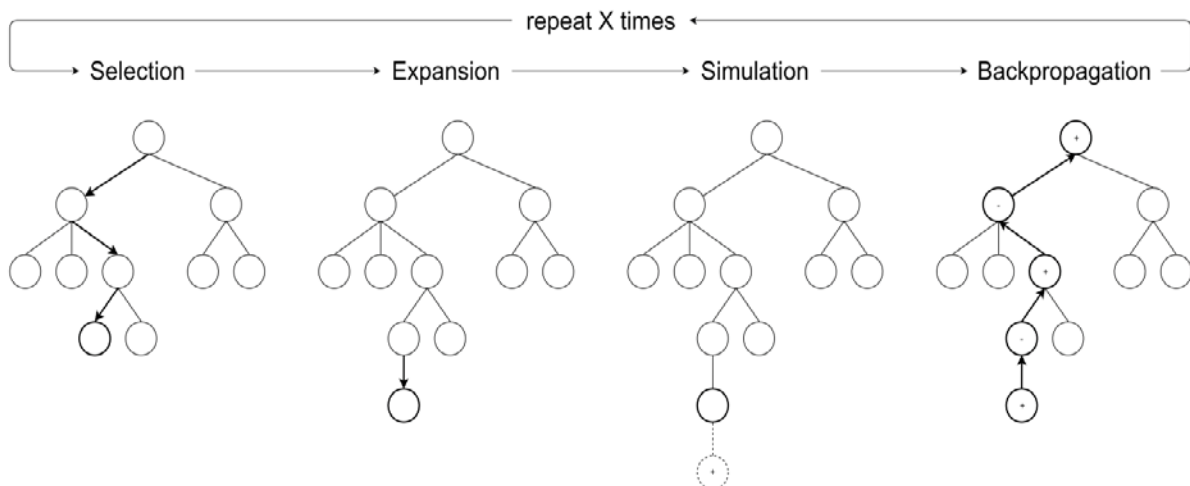


рис.1 Схема алгоритма MCTS

Каждый узел в дереве решений, помимо указателей на дочерние узлы, содержит следующую информацию:

- количество посещений данного узла алгоритмом;
- количество успешных симуляций с использованием данного узла;
- информацию о ходе.

Для правильного выбора узла на шаге «Selection» в MOTiFS также применяется метод Upper Confidence Bound applied to Trees (UCT), но несколько модифицированный.

$$UCT_{v,p} = \max(Q_v) + C \times \sqrt{\frac{2 \times \ln(N_p)}{N_v}},$$

где v – выбранный узел, p – родительский узел для выбранного узла, N_v и N_p – количество посещений выбранного и родительского узлов соответственно, $\max(Q_v)$ – максимальная награда в узле v (значением Q_v является отношение количества верно классифицированных объектов к общему количеству объектов из обучающей выборки и устанавливается на этапе обратного распространения – backpropagation), C – константа, выбираемая экспериментально.

Основными параметрами алгоритма являются константа C , применяемая для модифицированного метода *UCT*, и количество итераций для работы алгоритма. Чаще всего в качестве значений C выбирают числа 0.1, 0.05, 0.02. Количество итераций так же выбирается чаще всего из 500, 1000 или 10000 итераций.

Метод MOTiFS имеет вычислительную сложность равную $O(nb+c)$, где n – количество признаков, b – сложность операции выбора узла в алгоритме MCTS и c – сложность алгоритма классификации. Но так как сложность операции выбора узла является константной во время работы алгоритма, то вычислительная сложность может быть представлена как $O(sn+sc)$, где s – количество итераций симуляции.

Расширяемость алгоритма является одной из его положительных сторон. Предложены несколько различных модификаций алгоритма, позволяющих уменьшить время поиска решения. Алгоритм может быть реализован с использованием данных об области применения, что позволит производить более правдоподобные симуляции и снизить время поиска, но при этом возможно снижение производительности алгоритма (например, если используются более подробные симуляции) [6].

При реализации программного обеспечения были использованы следующие парадигмы программирования:

- объектно-ориентированная парадигма программирования, которая позволяет удобно описывать объекты и их признаки;
- рефлексивная парадигма программирования, с помощью которой возможна разработка методов классификации и отбора признаков без привязки к конкретным типам объектов и их признакам.

Для программного обеспечения была выбрана модульная архитектура, что позволяет беспрепятственно расширять количество обрабатываемых наборов данных и применимые к этим данным алгоритмы классификации и критерии качества классификации.

Программное обеспечение для метода MOTiFS было реализовано на языке C# с использованием платформы .NETCore 3.1. Данный выбор был обоснован тем, что C# и .NETCore предоставляют широкий ряд возможностей для указанных выше парадигм программирования без существенных проблем с производительностью.

Для разработки была использована среда разработки JetBrains Rider 2020.1.3, обладающая широкими возможностями статического анализа кода и его тестирования.

Разработанное программное обеспечение LocMOTiFS состоит из четырех модулей, что позволяет удобно его расширять:

- базовый модуль, включающий классы для объектов с признаками;
- модуль с методами классификации;
- модуль для работы с наборами данных;
- модуль для алгоритма отбора признаков MOTiFS.

Сравнительное тестирование производилось на оборудовании со следующими характеристиками: процессор Intel Core i7 7500U, ОЗУ 16 гигабайт, ОС Microsoft Windows 10 Pro.

Тренировочные наборы были взяты из двух публичных хранилищ, LIBSVM [1] и UCI [7]. Список используемых тренировочных наборов приведен в таблице 1. Указанные

наборы данных являются наиболее популярными и распространенными для тестирования методов классификации, регрессии и отбора признаков.

Базовая задача – решение общей задачи классификации (число классов от двух и более), осуществляемое на указанных наборах данных (см. таблицу 1).

Таблица 1.
Список используемых тренировочных наборов данных

№	Набор данных	Количество признаков	Количество образцов	Количество классов	Исходная точность классификации
1	Spambase	57	4701	2	0.673
2	Glass	9	214	6	0.703
3	DNA	180	2000	2	0.685
4	Waveform	40	5000	3	0.756
5	Breast cancer	10	683	2	0.845
6	Ionosphere	34	351	2	0.801
7	Wine	13	178	3	0.795
8	German credit data	24	1000	2	0.576
9	KlekotaRothfingerprint	4860	215	2	0.415
10	Multiple features	649	2000	10	0.657

Разработанное программное обеспечение LocMOTiFS использовалось для тестирования метода MOTiFS. Для сравнения с другими методами снижения размерности использовалось ПО wekalibrary [8].

Тестирование метода MOTiFS проводилось в сравнении с различными методами направлений feature extraction и feature selection (в основном методы обертки), а также с известными к настоящему времени современными методами, использующими MCTS (FUSE и MCTS-Relief). В таблице 2 приведены эти методы.

Таблица 2.
Список сравниваемых методов

Аббревиатура	Метод	Класс метода
PCA	Метод главных компонент	Метод выделения
FA	Факторный анализ	Метод выделения
MS	Многомерное шкалирование	Метод выделения
LDA	Линейный дискриминантный анализ	Метод выделения
IG	Information Gain	Метод фильтрации
SFS	Прямой поиск	Метод обертки
SBS	Обратный поиск	Метод обертки
FS-FS	Метод подобия признаков	Метод обертки
SFSW	Метод одновременного отбора признаков и взвешивания	Метод обертки
FUSE	Метод Feature UCT Selection	Метод обертки
MCTS-Relief	Метод MCTS-Relief	Метод обертки

.В качестве алгоритма классификации был использовался метод «k-ближайших соседей». Для оценки критерия качества классификации использовались средняя точность классификации (ассигасу) и среднее количество отобранных признаков.

Следует отметить, что не все методы возможно применить к определенным наборам данных. Это связано с тем, что некоторые методы не могут произвести отбор за адекватное время, либо количество признаков превышает допустимое количество для некоторого алгоритма отбора признаков. Результаты для таких методов отмечены как «не просчитано» (н/п).

Учитывая ограниченный объем тезисов, здесь для иллюстрации в табличной форме представлены результаты вычислений только по 4 наборам данных: Spambase - 57 пр. 2 класса, Wine – 13 пр. 3 класса, Class – 9 пр. 6 классов, Multiple features – 649 пр. 10 классов. В таблицах указаны точность классификации, количество полученных признаков и время вычисления каждым из рассмотренных методов.

Таблица 3.

Результаты сравнительного тестирования MOTiFS с другими методами

Набор данных	MOTiFS	PCA	FA	MS	LDA	IG	SFS	SBS
Spambase (57 пр)	0.907 31 пр. 43 мин.	0.687 12 пр. 10 мин.	0.751 15 пр. 8 мин.	0.901 10 пр. 9 мин.	0.649 16 пр. 10 мин.	0.845 24 пр. 17 мин.	0.874 35 пр. 12 мин.	0.874 35 пр. 12 мин.
Glass (9 пр)	0.705 5 пр. 2 мин.	0.701 4 пр. 1 мин.	0.698 6 пр. 1 мин.	0.703 5 пр. 1 мин.	0.708 5 пр. 1 мин.	0.681 4 пр. 1 мин.	0.631 6 пр. 34 сек	0.636 7 пр. 45 сек
Ionosphere (34 пр)	0.889 12 пр. 6 мин.	0.799 32 пр. 1 мин.	0.735 18 пр. 2 мин.	0.589 22 пр. 2 мин.	0.612 31 пр. 1 мин.	0.725 30 пр. 2 мин.	0.887 1 пр. 4 мин.	0.859 9 пр. 4 мин.
Multiplefeatures (649 пр)	0.980 321 пр. 37 мин.	0.775 491 пр. 7 мин.	0.533 64 пр. 11 мин.	0.815 169 пр. 1 мин.	0.812 457 пр. 14 мин.	0.729 627 пр. 9 мин.	0.903 210 пр. 18 мин.	0.912 305 пр. 17 мин.

Таблица 3 (продолжение).

Результаты сравнительного тестирования MOTiFS с другими методами

Набор данных	MOTiFS	FS-FS	SFSW	FUSE	MCTS-Relief
Spambase (57 пр)	0.907 31 пр. 43 мин.	0.900 29 пр. 15 мин.	0.907 31 пр. 27 мин.	н/п	0.898 34 пр. 47 мин.
Glass (9 пр)	0.705 5 пр.	н/п	0.678 4 пр.	0.701 6 пр.	0.702 5 пр.

	2 мин.		58 сек.	3 мин.	2 мин.
Ionosphere (34 пр)	0.889 12 пр. 6 мин.	0.788 16 пр. 6 мин.	0.883 11 пр. 2 мин.	н/п	0.857 9 пр. 7 мин.
Multiple features (649 пр)	0.980 321 пр. 37 мин.	н/п	0.979 270 пр. 29 мин.	н/п	0.971 274 пр. 46 мин.

Проведенное сравнительно тестирование показывает, что новый метод MOTiFS, реализованный в виде LocMOTiFS, во многих случаях производит более качественный (с позиций классификации) отбор признаков в сравнении с известными методами отбора признаков, включая и существующие современные методы, использующие алгоритм MCTS, превосходя последние в некоторых случаях и по времени вычисления.

В качестве недостатка следует отметить относительно низкую скорость вычислений LocMOTiFS, связанную с постоянным обходом дерева решений, присущий, кстати, и методам FUSE и MCTS-Relief, также основанным на MCTS. Возможно, сказалась и недостаточная оптимизация разработанного ПО.

Тем не менее, MOTiFS и его программная реализация LocMOTiFS показали свою перспективность использования во многих прикладных задачах, связанных с уменьшением размерности признакового пространства.

Список литературы

1. LIBSVM Data: Classification, Regression and Multi-label [Электронный ресурс]/ <https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/>
2. Alpaydm E. Introduction to Machine Learning. Massachusetts : MIT, 2010. – 579 p.
3. Muhammad Habib ur Rehman and etc Big Data Reduction Methods: A Survey/ Data Sci. Eng. (2016) 1: pp.265–284
4. Chaudhry M., Lee J. MOTiFS: Monte Carlo Tree Search Based Feature Selection // MDPI Entropy. – 2018, 20 (5), 385
5. Monte Carlo Tree Search in Go pp. 1-32. [Электронный ресурс] // <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.699.1458&rep=rep1&type=pdf>
6. Поиск по дереву методом Монте-Карло и крестики-нолики [Электронный ресурс]//Хабр – Режим доступа: <https://habr.com/ru/post/330092/>
<https://archive.ics.uci.edu/ml/datasets.php>
7. UCI Machine Learning Repository [Электронный ресурс]/ <https://archive.ics.uci.edu/ml/datasets.php>
8. Weka Library repository [Электронный ресурс] // University of Waikato – Режим доступа: <https://www.cs.waikato.ac.nz/~ml/weka/>

СТЕГОАНАЛИЗА АУДИОФАЙЛОВ ФОРМАТА WAVE

Б.В.Олейников¹, А.В.Стрельников²

¹Сибирский федеральный университет,

Институт Математики и Фундаментальной Информатики, **oleynik48@mail.ru**

² Сибирский федеральный университет,

Институт Математики и Фундаментальной Информатики, avstrelnikov95@gmail.com

В настоящее время в связи с активным развитием различных методов стеганографии все более актуальной становится задача стегоанализа — обнаружения скрытых вложений в мультимедиа файлы. Одними из наиболее емких стегоконтейнеров являются контейнеры на базе аудиофайлов формата WAVE [1]. Поэтому в дальнейшем основное внимание будет уделено им.

Современные подходы к стегоанализу строятся на использовании методов машинного обучения. Эти методы обучаются на данных, представленных в различных многомерных признаковых пространствах [2]. В свою очередь, признаковые пространства могут порождаться различными способами, к примеру, пространства, порожденные статистическими характеристиками дискретных вейвлет-преобразований (ДВП) аудиосигнала [3], энергии и энтропии аудиосигнала [4], производной аудиосигнала [5] и прочих.

Статистические характеристики дискретного вейвлет-преобразования являются эффективным инструментом для задачи анализа аудиофайлов [3, 6]. Также, в различных задачах анализа аудио, не связанных со стегоанализом, довольно часто в качестве признакового пространства используются характеристики пакетного вейвлет-преобразования (ПВП) [7]. В то же время, авторами не обнаружено работ, посвященных использованию пакетного вейвлет-преобразования для стегоанализа аудио.

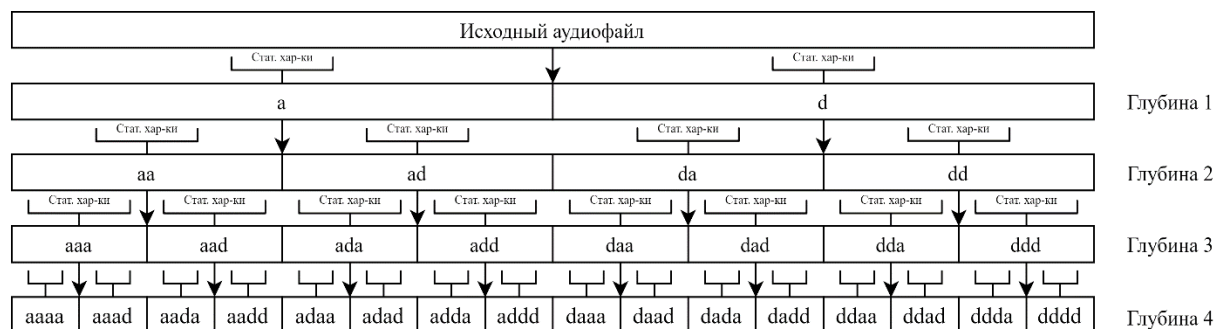


Рис. 1. Схема получения коэффициентов ПВП до глубины 4 для построения признакового пространства

В настоящей работе в качестве признакового пространства рассматриваются статистические характеристики (среднее, среднеквадратическое отклонение, коэффициенты асимметрии и эксцесса, максимум и минимум) последовательностей пакетного вейвлет-преобразования Добеши db8 [8] глубины 8 аудиофайлов. В отличие от ДВП, ПВП аудиофайла содержит большее количество подпоследовательностей, содержащих наименее значимую информацию об аудиосигнале [7]. Но, как известно, большинство методов стеганографии аудиофайлов используют как раз малозначимые

области (к примеру, младшие биты) для встраивания информации [1]. В связи с этим в данной работе предполагается, что с помощью признаков, построенных на таких подпоследовательностях, можно будет с приемлемой точностью отличать аудиофайлы с вложениями от аудиофайлов без вложений. В качестве примера на рис.1 представлена схема получения аппроксимирующих и детализирующих коэффициентов ПВП до глубины 4, на статистических характеристиках которых строится признаковое пространство.

Можно предположить также, что не все получаемые таким образом признаки будут одинаково информативны с позиции стегоанализа. Какая-то часть признаков может не влиять на качество классификации, а какая-то часть даже может вносить шумы, приводящие к ошибкам первого или второго рода. В целях выделения наиболее информативных признаков с позиции стегоанализа представляется целесообразным проведение снижения размерности признакового пространства. При этом сокращение признакового пространства также может привести и к снижению времени как на вычисление характеристик признаков, так и на скорость обучения классификатора.

В целом методы снижения размерности признакового пространства делятся на два класса: методы выделения признаков (feature extraction) и методы отбора признаков (feature selection). Методы отбора признаков, в свою очередь, подразделяются на методы фильтрации (filter), обертки (wrapper) и встраивания (embedded). В настоящей работе в целях снижения размерности признакового пространства используются как методы выделения, так и методы отбора признаков, и в дальнейшем производится их качественное сравнение с позиции стегоанализа.

Из каждого класса методов снижения размерности признакового пространства были взяты методы, наиболее часто используемые для решения практических задач и дающие в среднем лучший результат по разным выборкам. В качестве методов выделения признаков были рассмотрены факторный анализ (FA), метод главных компонент (PCA), сингулярное разложение (SVD) и быстрый анализ независимых компонент (Fast ICA). В качестве методов фильтрации были рассмотрены методы, основанные на максимальном коэффициенте информации (MIC). В качестве метода обертки был рассмотрен метод рекурсивного исключения признаков (RFE — Recursive Feature Elimination). Методы отбора признаков, относящихся к группе встраиваемых, не рассматривались.

Выделение признаков с помощью метода главных компонент происходило с назначенным выделением 10 (PCA 10) и 50 (PCA 50) главных компонент. С помощью факторного анализа выделялось 10 признаков. С помощью сингулярного разложения выделялось 10 (SVD 10) и 50 (SVD 50) признаков. С помощью алгоритма Fast ICA выделялось 10 признаков. Реализация данных методов была взята из библиотеки Scikit-learn [9].

Отбор признаков на основе MIC проводился несколькими способами. В первом был рассчитан MIC каждого признака с меткой класса; из исходного набора признаков были отсеяны признаки с MIC менее 0.3, данный метод далее в работе имеет название

MIC filter. Также был применен алгоритм отбора признаков на основе MIC, изложенный в работе [10]. Данный алгоритм отбирал 5 (MIC 5), 20 (MIC 20) и 40 (MIC 40) признаков.

Поскольку признаковое пространство обладает достаточно большой размерностью, метод рекурсивного исключения признаков RFE применялся совместно с методом фильтрации MIC filter. Метод RFE отбирал 5, 10 и 20 признаков; далее в работе эти методы названы как MIC RFE 5, MIC RFE 10 и MIC RFE 20, соответственно.

В качестве исходной выборки аудиофайлов был использован набор данных RAVDESS Emotional Speech [11]. Для унификации все файлы были обрезаны до двух секунд; файлы, длина которых не достигала 2 секунд, были исключены из набора. Таким образом, исходный файловый набор, представляющий собой пустые стегоконтейнеры (далее Empty), составил 1420 16-битных аудиофайлов формата WAVE с одним каналом и частотой дискретизации 48 КГц. Во все файлы набора Empty были произведены вложения случайной информации методом LSB с использованием ПО DeepSound [12], дающим выходные файлы разного качества: Low (вложение в 8 младших бит), Normal (в 4 младших бита) и High (в 2 младших бита) на 25, 50, 75 и 100 процентов от максимальной возможности стегоконтейнера, определяемой DeepSound. Пакет DeepSound был выбран как один из самых известных за счет повсеместно встречающихся руководств по его использованию [13]. Вложения происходили в автоматическом режиме с помощью разработанного авторами данной статьи программного обеспечения PyAutoEmbedDS на языке Python с использованием сторонней библиотеки PyAutoGUI. В конечном итоге, файловая выборка составила 18460 аудиофайлов: по 1420 файлов в наборах Empty, High25, High50, High75, High100, Normal25, Normal50, Normal75, Normal100, Low25, Low50, Low75 и Low100.

Для каждого из аудиофайлов было рассчитано признаковое пространство, основанное на статистических характеристиках пакетного вейвлет-преобразования Добеши db8 глубины 8 с использованием библиотек PyWavelets [14] и NumPy [15]. В итоге исходное признаковое пространство составило 3060 признаков.

В качестве классификаторов использовались: логистическая регрессия (LogReg), метод k ближайших соседей (KNN), линейный метод опорных векторов (SVC), дерево решений (DesTree), градиентный бустинг деревьев решений (GradBoost) и случайный лес (RanFor). Качество классификации оценивалось с помощью accuracy (правильности классификации). Реализация данных методов была взята из библиотеки Scikit-learn [9] для языка программирования Python.

Разделение полученного набора данных на тренировочную и тестовую выборку происходило следующим образом. Для получения равновесной тренировочной выборки из набора Empty бралось 1200 файлов (относятся к классу «0»), из остальных наборов (Low, Normal и High) последовательно бралось по 100 файлов. Так, в наборе Low25 были взяты файлы 1-100, в Low50 101-200, в Low75 201-300, в Low100 301-400, Normal25 401-500, Normal50 501-600, Normal75 601-700, Normal100 701-800, High25 801-900, High50 901-1000, High75 1001-1100, High100 1101-1200, итого также 1200 (относящихся к классу «1»). Оставшиеся файлы составили тестовую выборку, которая в итоге получилась неравновесной. Отметим, что в дальнейшем тестирование производилось как на

исходной неравновесной выборке, так и на равновесной, полученной при использовании oversampling, и дало одинаковые результаты.

Все вышеописанные методы снижения размерности признакового пространства были протестированы при применении указанных шести классификаторов. В таблице представлены лучшие результаты правильности классификации среди всех рассмотренных методов машинного обучения на тестовой выборке. Значения в таблице отсортированы по убыванию средней правильности.

Таблица 1

Значения правильности бинарной классификации на выделенных наборах признаков

Метод отбора признаков	Кол-во отобранных х признаков	Лучший ML метод	Класс "0" (Empty)	Класс "1"	Отдельно по наборам для класса "1"			Средняя правильность
					High	Normal	Low	
Нет	3060	RanFor	0.9000	0.9998	0.9994	1.0000	1.0000	0.9921
MIC RFE 20	20	RanFor	0.9091	0.9977	0.9932	1.0000	1.0000	0.9909
MIC filter	43	RanFor	0.9045	0.9979	0.9938	1.0000	1.0000	0.9907
MIC RFE 10	10	RanFor	0.9136	0.9953	0.9858	1.0000	1.0000	0.9890
MIC RFE 5	5	RanFor	0.9091	0.9953	0.9864	0.9994	1.0000	0.9886
MIC 20	20	RanFor	0.8591	0.9977	0.9930	1.0000	1.0000	0.9870
MIC 40	40	RanFor	0.8591	0.9972	0.9917	1.0000	1.0000	0.9866
MIC 5	5	RanFor	0.8682	0.9955	0.9866	1.0000	1.0000	0.9857
FA	10	DesTree	0.7273	0.8057	0.5148	0.9051	0.9973	0.7997
SVD 50	50	LogReg	0.5545	0.4642	0.3902	0.3801	0.6223	0.4712
PCA 50	50	LogReg	0.5364	0.4571	0.3782	0.3670	0.6261	0.4632
SVD 10	10	KNN	0.4682	0.4362	0.2339	0.3750	0.6998	0.4387
PCA 10	10	LogReg	0.5091	0.4213	0.4045	0.3953	0.4640	0.4280
Fast ICA	10	LogReg	0.5091	0.4206	0.4036	0.3943	0.4638	0.4274

Среди всех методов снижения размерности наилучшим по правильности классификации оказался метод MIC RFE 20 с 20 признаками, используемыми классификатором «Случайный лес» (RanFor) со средним качеством классификации в 0.9909.

Правильность классификации на получаемом при таком подходе признаковом пространстве чуть ниже, чем на исходном. Однако, средняя правильность понизилась лишь на 0.001, а количество признаков сократилось на 3040 (в 153 раза). Это позволяет значительно снизить количество требуемой памяти для хранения признаков, повысить скорость обучения классификаторов, а также снизить время на вычисление характеристик признаков для аудиофайлов с практически незначительной потерей точности.

Результаты правильности для всех рассмотренных классификаторов для метода MIC RFE 20 представлены в виде гистограммы на рис. 2.

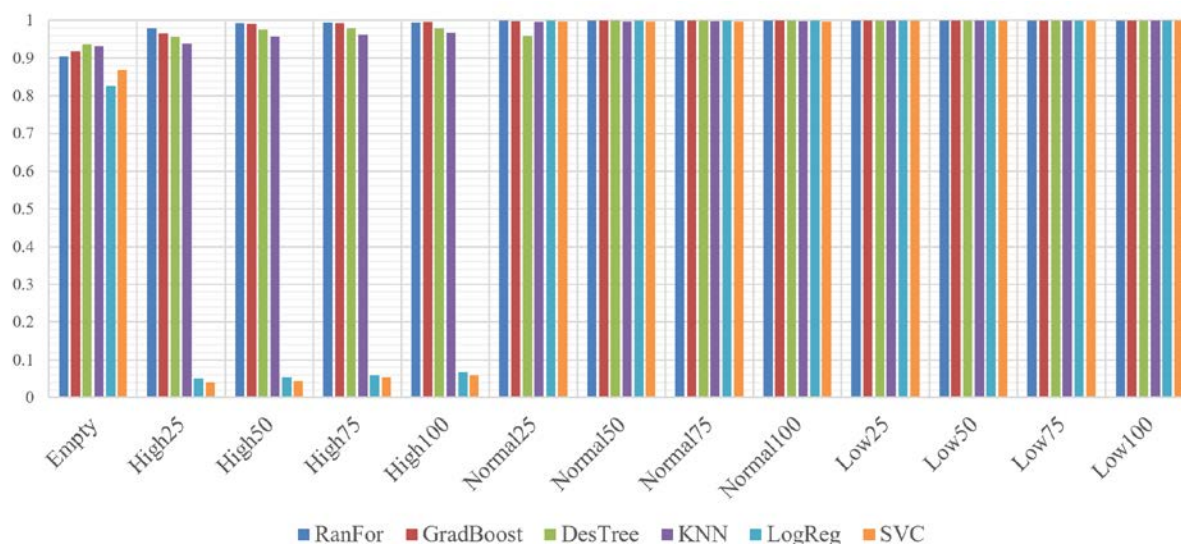


Рис.2. Правильность классификации на тестовой выборке, включающей различные наборы данных, в зависимости от метода машинного обучения для метода MIC RFE 20

Гистограмма показывает, что для всех файлов Normal и Low, независимо от процента заполнения контейнера, практически все 6 рассмотренных классификаторов одинаково успешно обнаруживают вложения со 100% точностью. В то же время для файлов High ни один из классификаторов не достигает 100% точности обнаружения, а некоторые из них (LogReg, SVC) и, вообще, работают плохо. Это можно использовать при первичном стегоанализе аудиофайлов.

С помощью выявленного лучшего метода (MIC RFE 20) среди 3060 исходных признаков, полученных на основе статистических характеристик ПВП Добеши db8 глубины 8, были отобраны следующие 20 признаков, касающихся аппроксимирующих (a) и детализирующих (d) коэффициентов на глубинах от 5 до 8: 'daaaaaad_std', 'daaaaaada_std', 'daaaaaadd_std', 'daaaaaaad_std', 'daaaaadda_std', 'daaaaadd_std', 'daaaaaada_min', 'daaaaaad_min', 'daaaaa_std', 'daaaaad_std', 'daaaaaddd_std', 'daaaaada_std', 'daaaaaad_max', 'daaaaaad_min', 'daaaaadaa_std', 'daaaaa_max', 'daaaaa_min', 'daaaaa_max', 'daaaaa_min', 'daaaaaa_min'. Как видно из перечисленного набора, по большей части наиболее информативными являются среднеквадратическое отклонение (std), а также максимум (max) и минимум (min) аппроксимирующих и детализирующих коэффициентов, рассчитанных от детализирующих коэффициентов глубины 1, и имеющих важное значение для задач стегоанализа начиная с глубины 5. Это говорит об особой значимости учета статистических характеристик второстепенных коэффициентов ПВП в решении задачи стегоанализа аудиофайлов.

Основным результатом работы стало получение приемлемого признакового пространства, с помощью которого возможно отличать аудиофайлы формата WAVE со стеганографическими вложениями методом LSB и без.

Подходы, рассмотренные в данной работе, могут быть использованы для стегоанализа аудиофайлов, а также для решения различных задач, связанных с классификацией аудио- и других цифровых дискретных сигналов.

Список литературы

1. Singh P. A comparative study of audio steganography techniques //International Research Journal of Engineering and Technology (IRJET). – 2016. – V. 3. – №. 4. – pp. 580-585.
2. Ghasemzadeh H., Kayvanrad M. H. Comprehensive review of audio steganalysis methods //IET Signal Processing. – 2018. – V. 12. – №. 6. – pp. 673-687
3. Ru X. M., Zhang H. J., Huang X. Steganalysis of audio: Attacking the steghide //2005 International Conference on Machine Learning and Cybernetics. – IEEE, 2005. – V. 7. – pp. 3937-3942.
4. Djebbar F., Ayad B. Energy and entropy based features for WAV audio steganalysis //Journal of Information Hiding and Multimedia Signal Processing. – 2017. – V. 8. – №. 1. – pp. 168-181.
5. Liu Q., Sung A. H., Qiao M. Derivative-based audio steganalysis //ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM). – 2011. – V. 7. – №. 3. – pp. 1-19.
6. Олейников Б.В., Стрельников А.В. Подход к определению эффективных признаков для бинарной классификации аудиофайлов с использованием дискретного вейвлет-преобразования //Приоритетные направления инновационной деятельности в промышленности. Сборник научных статей по итогам второй международной научной конференции. – 2020. – С. 35-38.
7. Бердибаева Г. К., Бодин О. Н., Фирсов Д. С. Классификация звуков астматического дыхания с использованием нейронных сетей //Измерение. Мониторинг. Управление. Контроль. – 2018. – №. 2 (24). – с. 86-90.
8. Daubechies I. Ten lectures on wavelets. – Society for industrial and applied mathematics, 1992.
9. Pedregosa F. et al. Scikit-learn: Machine learning in Python //the Journal of machine Learning research. – 2011. – V. 12. – pp. 2825-2830.
10. Chen Z. et al. Combining MIC feature selection and feature-based MSPCA for network traffic anomaly detection //2016 Third International Conference on Digital Information Processing, Data Mining, and Wireless Communi-cations (DIPDMWC). – IEEE, 2016. – pp. 176-181.
11. Livingstone S. R., Russo F. A. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAV-DESS): A dynamic, multimodal set of facial and vocal expressions in North American English //PloS one. – 2018. – V. 13. – №. 5. – . e0196391.
12. DeepSound. [Электронный ресурс] : Описание ПО и ссылка на скачивание. – 2015. – Словения. – Режим доступа: <http://jpinsoft.net/DeepSound/Overview.aspx>
13. Прячем секретные файлы внутри аудиозаписи как Мистер Робот (DeepSound). [Электронный ресурс] : Пикабу. – 2018. – Россия. – Режим доступа: https://pikabu.ru/story/pryachem_sekretnyye_faylyi_vnutri_audiozapisi_kak_mister_robot_deepsound_5854241
14. Lee G. R. et al. PyWavelets: A Python package for wavelet analysis //Journal of Open Source Software. – 2019. – V. 4. – №. 36. – pp. 1237-1238.
15. Travis E. Oliphant. A guide to NumPy, USA: Trelgol Publishing, 2003. — 378 p.

КЛАСТЕРИЗАЦИЯ ПОСРЕДСТВОМ КВАНТОВОГО ОТЖИГА НА КВАНТОВЫХ ТРЁХУРОВНЕВЫХ ЭЛЕМЕНТАХ - КУТРИТАХ

И. С. Пичковский¹, В. Е. Зобов¹.

¹Институт физики им. Л.В. Киренского Сибирского отделения Российской академии наук - обособленное подразделение ФИЦ КНЦ СО РАН

В настоящее время усилия многих учёных направлены на создание квантового компьютера, который теоретически должен превосходить классический компьютер. В том числе и на создание квантовых нейронных сетей, одной из задач которых является задача кластеризации (разбиение множества на подмножества (группы)). Начиная от полуклассического подхода [1] и заканчивая полностью квантовыми алгоритмами [2].

Большинство исследователей по аналогии с классическими нейронами, имеющими два состояния, используют в качестве квантовых нейронов кубиты – квантовые двухуровневые элементы. И поэтому, чаще всего, рассматривают разбиение на две группы. В общем случае каждой точке ставится в соответствие то количество кубитов, что и количество групп [2]. В нашей работе мы будем ориентироваться на NV центры в алмазе, обладающие интересными для создания квантового компьютера свойствами: оптическое считывание и инициация, а также возможностью расщеплять резонансные частоты ЭПР с помощью градиента механического напряжения и магнитного поля [3]. Если рассматривать энергетическую структуру NV-центра, то он имеет два уровня энергии, каждый из которых имеет тонкую структуру, которая делит основное состояние на три уровня, разделенные между собой энергетической щелью СВЧ диапазона. Именно это позволяет рассматривать квантовые нейронные сети на трёхуровневых квантовых элементах – кутритах, взятых в качестве квантовых нейронов. Так же из-за этого мы в первую очередь рассмотрели разбиение множества на три группы.

Наша задача кластеризации решалась посредством адиабатической эволюции во времени эффективного гамильтониана [4]:

$$H(t) = \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_p, \quad (1)$$

где $H_0 = h \sum_{i=1}^n S_i^x$ - начальный гамильтониан взаимодействия с поперечным магнитным

полем, основное состояние которого легко приготовить, а H_p - целевой гамильтониан, основное состояние которого кодирует решение нашей задачи.

В литературе [2] существуют два метода кластеризации: one hot encoding и k-средних. В нашей работе в качестве данных будут использованы точки в двумерном декартовом пространстве, полученные с помощью генератора случайных чисел.

Для дальнейшего решения таких задач введём кодировку в собственных состояниях проекций спина на ось Z : $|1\rangle$, $|0\rangle$ и $|-1\rangle$. Для начала решим задачу разбиения на три группы шести точек с координатами (4,-2), (-7,7), (6,-9), (-6,8), (-2,-6), (-9,5) - с помощью метода one hot encoding. В этом методе каждой точке ставится в соответствие оператор спина S_i^z в гамильтониане H_p , где i - номер точки. Одной из точек

присваивают фиксированное априорное значение (в нашем случае первому спину присваиваем значение $S_1^z = 1$), а состояния остальных спинов m_i определяются минимальной энергией гамильтониана H_p в результате адиабатической эволюции (1). Точки с одинаковыми состояниями спинов относятся к одной группе.

Гамильтониан взаимодействия между двумя точками для решения нашей задачи возьмём в следующем виде:

$$H_{\text{int}} = R_{ij} \left(2 \left[|1,1\rangle\langle 1,1|_{i,j} + |0,0\rangle\langle 0,0|_{i,j} + |-1,-1\rangle\langle -1,-1|_{i,j} \right] - 1 \right), \quad (2)$$

где R_{ij} - расстояние между точками i и j , определяемое по следующей формуле:

$$R_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, \quad (3)$$

где x_i, x_j, y_i и y_j координаты точек на декартовой плоскости, а $|m_i, m_j\rangle\langle m_i, m_j|_{i,j}$ - проектор на двухспиновое состояние точек i и j . После чего целевой гамильтониан для решения нашей задачи примет следующий вид:

$$\begin{aligned} H_p = & \sqrt{202} \left[2 \left(|1,1\rangle\langle 1,1|_{1,2} + |0,0\rangle\langle 0,0|_{1,2} + |-1,-1\rangle\langle -1,-1|_{1,2} \right) - 1 \right] + \\ & + \sqrt{53} \left[2 \left(|1,1\rangle\langle 1,1|_{1,3} + |0,0\rangle\langle 0,0|_{1,3} + |-1,-1\rangle\langle -1,-1|_{1,3} \right) - 1 \right] + \\ & + \sqrt{200} \left[2 \left(|1,1\rangle\langle 1,1|_{1,4} + |0,0\rangle\langle 0,0|_{1,4} + |-1,-1\rangle\langle -1,-1|_{1,4} \right) - 1 \right] + \\ & + \sqrt{52} \left[2 \left(|1,1\rangle\langle 1,1|_{1,5} + |0,0\rangle\langle 0,0|_{1,5} + |-1,-1\rangle\langle -1,-1|_{1,5} \right) - 1 \right] + \\ & + \sqrt{218} \left[2 \left(|1,1\rangle\langle 1,1|_{1,6} + |0,0\rangle\langle 0,0|_{1,6} + |-1,-1\rangle\langle -1,-1|_{1,6} \right) - 1 \right] + \\ & + \sqrt{425} \left[2 \left(|1,1\rangle\langle 1,1|_{2,3} + |0,0\rangle\langle 0,0|_{2,3} + |-1,-1\rangle\langle -1,-1|_{2,3} \right) - 1 \right] + \\ & + \sqrt{2} \left[2 \left(|1,1\rangle\langle 1,1|_{2,4} + |0,0\rangle\langle 0,0|_{2,4} + |-1,-1\rangle\langle -1,-1|_{2,4} \right) - 1 \right] + \\ & + \sqrt{194} \left[2 \left(|1,1\rangle\langle 1,1|_{2,5} + |0,0\rangle\langle 0,0|_{2,5} + |-1,-1\rangle\langle -1,-1|_{2,5} \right) - 1 \right] + \\ & + \sqrt{8} \left[2 \left(|1,1\rangle\langle 1,1|_{2,6} + |0,0\rangle\langle 0,0|_{2,6} + |-1,-1\rangle\langle -1,-1|_{2,6} \right) - 1 \right] + \\ & + \sqrt{433} \left[2 \left(|1,1\rangle\langle 1,1|_{3,4} + |0,0\rangle\langle 0,0|_{3,4} + |-1,-1\rangle\langle -1,-1|_{3,4} \right) - 1 \right] + \\ & + \sqrt{73} \left[2 \left(|1,1\rangle\langle 1,1|_{3,5} + |0,0\rangle\langle 0,0|_{3,5} + |-1,-1\rangle\langle -1,-1|_{3,5} \right) - 1 \right] + \\ & + \sqrt{421} \left[2 \left(|1,1\rangle\langle 1,1|_{3,6} + |0,0\rangle\langle 0,0|_{3,6} + |-1,-1\rangle\langle -1,-1|_{3,6} \right) - 1 \right] + \\ & + \sqrt{212} \left[2 \left(|1,1\rangle\langle 1,1|_{4,5} + |0,0\rangle\langle 0,0|_{4,5} + |-1,-1\rangle\langle -1,-1|_{4,5} \right) - 1 \right] + \\ & + \sqrt{18} \left[2 \left(|1,1\rangle\langle 1,1|_{4,6} + |0,0\rangle\langle 0,0|_{4,6} + |-1,-1\rangle\langle -1,-1|_{4,6} \right) - 1 \right] + \\ & + \sqrt{170} \left[2 \left(|1,1\rangle\langle 1,1|_{5,6} + |0,0\rangle\langle 0,0|_{5,6} + |-1,-1\rangle\langle -1,-1|_{5,6} \right) - 1 \right] \end{aligned}$$

Результат решения данной задачи представлен на Рис.1, на котором группы выделены тремя цветами: синий - (-9,5), (-7,7) и (-6,8), красный - (-2,-6) и (4,-2) и

зелёный - (6, -9). Из Рис.1 видно, что наш алгоритм справился с решением данной задачи.

Существует более эффективный алгоритм, получивший название k-средних. В этом алгоритме точки данных группируются относительно центров масс, заданных вручную. Затем относительно этих групп вычисляется центр масс и расстояния от точек до центра масс, и снова разбиваются на группы. До тех пор пока результаты каждого вычисления не сойдутся к определённому распределению [5]. Мы воспользуемся модифицированным вариантом алгоритма – k-средних++ [6]. В нём в качестве центроид выбирают некоторое количество точек, равное числу групп, которым ставят в соответствие фиксированные значения проекций спинов. Гамильтониан взаимодействия точек с центроидами аналогичен гамильтониану (2). Рассмотрим в качестве примера кластеризацию на три группы следующих девяти точек: (8,-1), (-2,-6), (1,6), (4,-4), (3,8), (9,-4), (-5,8), (-6,-8), (3,-10). В качестве центроидов будем использовать точки: (8,-1), (-2,-6), (1,6). Целевой гамильтониан для решения данной задачи примет вид:

$$\begin{aligned}
 H_p = & \sqrt{25} \left[2 \left(|1,1\rangle\langle 1,1|_{1,1} + |0,0\rangle\langle 0,0|_{1,1} + |-1,-1\rangle\langle -1,-1|_{1,1} \right) - 1 \right] + \\
 & + \sqrt{106} \left[2 \left(|1,1\rangle\langle 1,1|_{1,2} + |0,0\rangle\langle 0,0|_{1,2} + |-1,-1\rangle\langle -1,-1|_{1,2} \right) - 1 \right] + \\
 & + \sqrt{10} \left[2 \left(|1,1\rangle\langle 1,1|_{1,3} + |0,0\rangle\langle 0,0|_{1,3} + |-1,-1\rangle\langle -1,-1|_{1,3} \right) - 1 \right] + \\
 & + \sqrt{250} \left[2 \left(|1,1\rangle\langle 1,1|_{1,4} + |0,0\rangle\langle 0,0|_{1,4} + |-1,-1\rangle\langle -1,-1|_{1,4} \right) - 1 \right] + \\
 & + \sqrt{245} \left[2 \left(|1,1\rangle\langle 1,1|_{1,5} + |0,0\rangle\langle 0,0|_{1,5} + |-1,-1\rangle\langle -1,-1|_{1,5} \right) - 1 \right] + \\
 & + \sqrt{106} \left[2 \left(|1,1\rangle\langle 1,1|_{1,6} + |0,0\rangle\langle 0,0|_{1,6} + |-1,-1\rangle\langle -1,-1|_{1,6} \right) - 1 \right] + \\
 & + \sqrt{40} \left[2 \left(|1,1\rangle\langle 1,1|_{2,1} + |0,0\rangle\langle 0,0|_{2,1} + |-1,-1\rangle\langle -1,-1|_{2,1} \right) - 1 \right] + \\
 & + \sqrt{221} \left[2 \left(|1,1\rangle\langle 1,1|_{2,2} + |0,0\rangle\langle 0,0|_{2,2} + |-1,-1\rangle\langle -1,-1|_{2,2} \right) - 1 \right] + \\
 & + \sqrt{125} \left[2 \left(|1,1\rangle\langle 1,1|_{2,3} + |0,0\rangle\langle 0,0|_{2,3} + |-1,-1\rangle\langle -1,-1|_{2,3} \right) - 1 \right] + \\
 & + \sqrt{205} \left[2 \left(|1,1\rangle\langle 1,1|_{2,4} + |0,0\rangle\langle 0,0|_{2,4} + |-1,-1\rangle\langle -1,-1|_{2,4} \right) - 1 \right] + \\
 & + \sqrt{20} \left[2 \left(|1,1\rangle\langle 1,1|_{2,5} + |0,0\rangle\langle 0,0|_{2,5} + |-1,-1\rangle\langle -1,-1|_{2,5} \right) - 1 \right] + \\
 & + \sqrt{41} \left[2 \left(|1,1\rangle\langle 1,1|_{2,6} + |0,0\rangle\langle 0,0|_{2,6} + |-1,-1\rangle\langle -1,-1|_{2,6} \right) - 1 \right] + \\
 & + \sqrt{109} \left[2 \left(|1,1\rangle\langle 1,1|_{3,1} + |0,0\rangle\langle 0,0|_{3,1} + |-1,-1\rangle\langle -1,-1|_{3,1} \right) - 1 \right] + \\
 & + \sqrt{8} \left[2 \left(|1,1\rangle\langle 1,1|_{3,2} + |0,0\rangle\langle 0,0|_{3,2} + |-1,-1\rangle\langle -1,-1|_{3,2} \right) - 1 \right] + \\
 & + \sqrt{164} \left[2 \left(|1,1\rangle\langle 1,1|_{3,3} + |0,0\rangle\langle 0,0|_{3,3} + |-1,-1\rangle\langle -1,-1|_{3,3} \right) - 1 \right] + \\
 & + \sqrt{40} \left[2 \left(|1,1\rangle\langle 1,1|_{3,4} + |0,0\rangle\langle 0,0|_{3,4} + |-1,-1\rangle\langle -1,-1|_{3,4} \right) - 1 \right] + \\
 & + \sqrt{245} \left[2 \left(|1,1\rangle\langle 1,1|_{3,5} + |0,0\rangle\langle 0,0|_{3,5} + |-1,-1\rangle\langle -1,-1|_{3,5} \right) - 1 \right] + \\
 & + \sqrt{260} \left[2 \left(|1,1\rangle\langle 1,1|_{3,6} + |0,0\rangle\langle 0,0|_{3,6} + |-1,-1\rangle\langle -1,-1|_{3,6} \right) - 1 \right]
 \end{aligned}$$

где члены $|n_i, m_j\rangle\langle n_i, m_j|_{i,j}$ - проекторы на двухспиновые состояние i центроида и j спина.

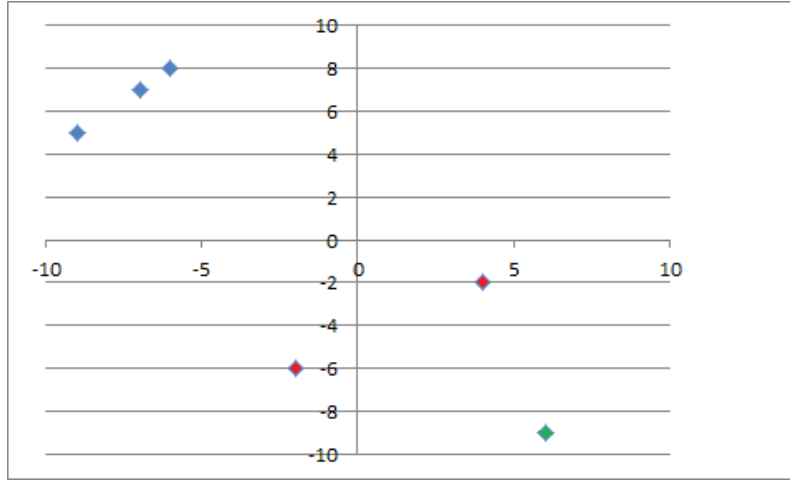


Рис.1. Результат разбиения множества на три группы методом one hot encoding.

Результаты кластеризации представлены на Рис.2. Синим выделены точки: $(-5,8)$, $(1,6)$, $(3,8)$; зелёным: $(-6,-8)$, $(-2,-6)$, $(3,-10)$ и красным: $(4,-4)$, $(8,-1)$ и $(9,-4)$. Как из рисунка видно, наш алгоритм справился с решением данной задачи.

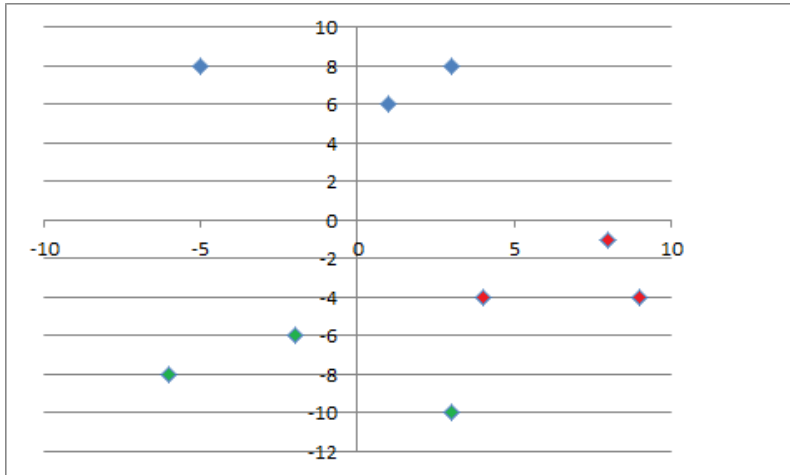


Рис.2. Результаты разбиения на три группы методом k-средних++.

Для полноты картины необходимо рассмотреть два других примера: разбиение на две группы и на четыре. Для начала рассмотрим кластеризацию на две группы методом one-hot encoding. Гамильтониан взаимодействия точек будет иметь тот же вид, что и в предыдущем случае (2), но потребуются дополнительно штрафовать все состояния с проекцией спина $S^z = -1$, а именно: $|1,-1\rangle$, $|0,-1\rangle$, $|-1,1\rangle$ и $|-1,0\rangle$. Для этого добавим к нему штрафной гамильтониан:

$$H_{fine} = R_{ij} (2[|1,-1\rangle\langle 1,-1|_{i,j} + |0,-1\rangle\langle 0,-1|_{i,j} + |-1,1\rangle\langle -1,1|_{i,j} + |-1,0\rangle\langle -1,0|_{i,j}] - 1) \quad (4)$$

Рассмотрим в качестве примера разбиение на две группы шести точек: $(6,6)$, $(-6,5)$, $(-3,9)$, $(4,-10)$, $(-7,4)$, $(-5,1)$ методом one hot encoding. Так же как и в предыдущем случае

использования данного метода, первой точке (6,6) ставим в соответствие фиксированный спин с априорным значением $S_1^z = 1$. Целевой гамильтониан примет вид:

[illegible]

Результат представлен на Рис.3. Как из него видно, алгоритм разбивает данные на две группы: красная - $(-7,4)$, $(-6,5)$, $(-5,-1)$ и $(-3,9)$, а также синяя группа $(4,-10)$ и $(6,6)$.

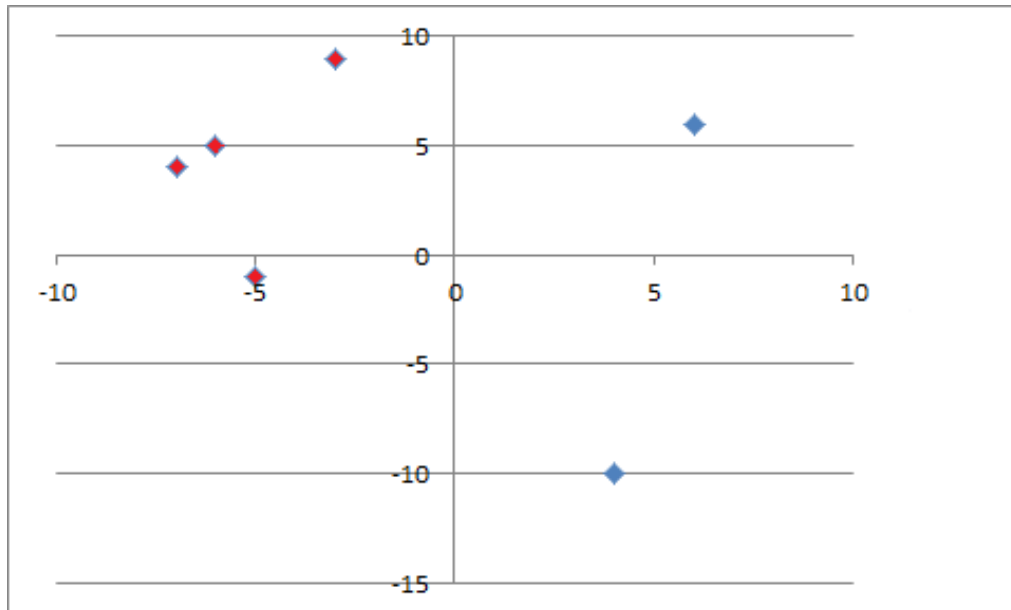


Рис.3. Результаты разбиения множества на две группы.

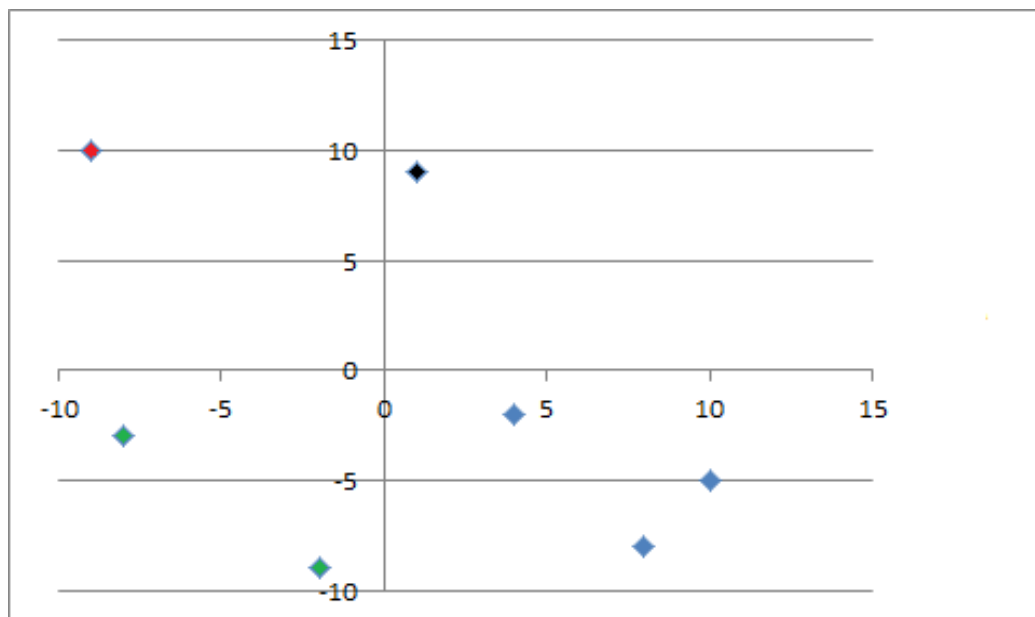


Рис.4. Результаты разбиения множества на четыре группы.

Наконец, рассмотрим последний случай - разбиение на четыре группы методом k -средних ++. Для этого мы каждую точку кодируем в состояниях двух спинов $|1,1\rangle$, $|1,0\rangle$, $|1,-1\rangle$ и $|0,1\rangle$. Теперь будем использовать оператор проектирования на состояние четырёх спинов $|m_1, m_2, n_1, n_2\rangle \langle m_1, m_2, n_1, n_2|$, где m_1 , m_2 - фиксированные проекции спинов, кодирующие центры, а n_1 и n_2 - проекции спинов, кодирующие остальные

точки. Поскольку мы из всего спектра, состоящему из 81 четырёхспиновых состояний, используем только 12, нам приходится дополнительно штрафовать 65 состояний. С помощью данного алгоритма было рассмотрено разбиение на четыре группы восьми точек $(-9,10)$, $(1,9)$, $(8,-3)$, $(-2,-9)$, $(4,-2)$, $(8,-8)$, $(10,-5)$, результат которого представлен на Рис.4. Из него видно, что алгоритм разбил на четыре группы: красная – $(-9,10)$, чёрная – $(1,9)$, зелёная: $(8,-3)$ и $(-2,-9)$ и синяя: $(4,-2)$, $(8,-8)$, $(10,-5)$. Аналогичным образом можно разбить на большее количество множеств.

Таким образом, было показано, что задача кластеризации может эффективно решаться на кутритах, при использовании меньшего числа элементов, чем при решении аналогичной задачи на кубитах.

Список литературы

1. Mendelson S. S. et al.: Quantum-Assisted Clustering Algorithms for NISQ-Era Devices. E-preprint: <https://arxiv.org/abs/1904.08992>.
2. Kumar V.: Quantum annealing for combinatorial clustering / V Kumar, G Bass, C Tomlin, J Dulny //Quantum Information Processing. – 2018. – Т. 17. – №. 2. – С. 39.
3. Xu Z.: Quantum information processing with closely-spaced diamond color centers in strain and magnetic fields / Z Xu, Z Yin, Q Han, T Li //Optical Materials Express. – 2019. – Т. 9. – №. 12. – С. 4654-4668.
4. Albash T.: Adiabatic quantum computation / T Albash, DA Lidar //Reviews of Modern Physics. – 2018. – Т. 90. – №. 1. – С. 015002.
5. Хант Э. Метод k средних / Э. Хант // В: Искусственный интеллект. Ред. В. Л. Стефанюка. - Москва: Мир - 1978.
6. Arthur D. Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms: k-means++: The advantages of careful seeding / D. Arthur, S. Vassilvitskii. - Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2007. - 1027

НЕЙРОСЕТЕВАЯ МОДЕЛЬ ДЛЯ УПРАВЛЕНИЯ ВЕРТИКАЛИЗАЦИЕЙ ЭКЗОСКЕЛЕТА В КОМБИНИРОВАННОМ РЕЖИМЕ ¹

А.А. Трифонов¹, А.А. Кузьмин¹, С.А. Филист¹, Е.В. Петрунина²

¹Юго-Западный государственный университет,

ФФиПИ, каф. биомедицинской инженерии, SFilist@gmail.com

²Московский государственный гуманитарно-экономический университет,
petrunina@mggeu.ru

Введение.

Для осуществления постинсультной реабилитации биотехническая система экзоскелет – пациент должна функционировать в комбинированном режиме. В определенный момент времени пациент занимает активное положение и выполняет заданное движение самостоятельно. В другой момент времени пациент занимает пассивное положение и экзоскелет осуществляет заданное программное движение или движение, определяемое посредством декодирования электромиограммы (ЭМГ). В этом случае пациент совершает заданные движения с помощью экзоскелета, а биотехническая система определяет момент и размер помощи [1, 3]. Поэтому декодер ЭМГ должен определить не только тип движения и координаты конечности, но и усилие (момент), которое передается на соответствующий исполнительный орган.

Несмотря на многочисленные исследования в области формирования пространства информативных признаков по поверхностным ЭМГ для классификаторов движений [2], проблема классификации ЭМГ-паттернов для осуществления комбинированного режима управления экзоскелетом пока не решена.

Методы.

Известно множество технических решений, позволяющих управлять экзоскелетом посредством биотехнической обратной связи [3, 5]. Для управления процедурой реабилитации необходимы модули принятия решений для расшифровки поверхностных ЭМГ. Обученные классификаторы используются в качестве таких модулей принятия решений. Они построены на основе моделей нейронных сетей, моделей нечеткого вывода или гибридных (гетерогенных) моделей [4, 8, 9]. В то же время обучающие выборки для обучения этих классификаторов формируются из базы данных, полученной в результате поискового анализа.

Структурная схема устройства управления экзоскелетом посредством электромиосигнала представлено на рис. 1.

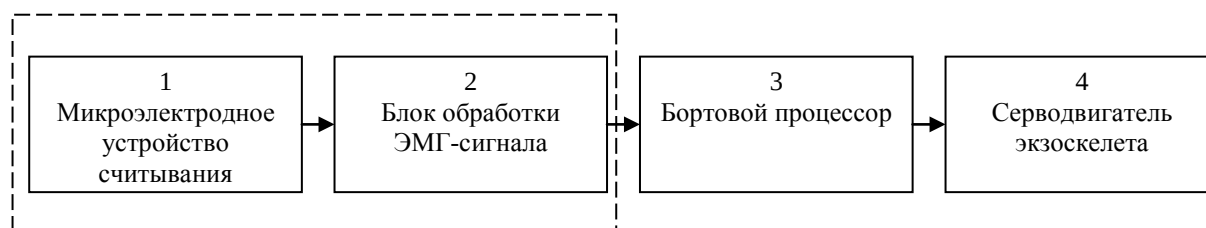


Рис. 1. Структурная схема устройства управления экзоскелетом посредством электромиосигнала

¹ Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 19-38-90112.

Acknowledgments: The reported study was funded by RFBR, project number 19-38-90112.

Устройство состоит из блока электродов 1, блока обработки ЭМГ – сигнала 2; бортового процессора 3, и контроллера серводвигателей 4. Отличительной особенностью этого устройства является отсутствие биотехнических обратных связей. Обратная связь осуществляется только через сенсорный канал оператора (пациента). Сущность предлагаемого метода управления экзоскелетом состоит в том, что при дешифрации ЭМГ используется не только ее амплитудные показатели, но и частотные характеристики, так как известно, что повышение двигательной активности приводит не только к увеличению амплитуды электромиосигнала, но и к увеличению числа задействованных двигательных единиц. То есть для кодирования ЭМГ используется гипотеза об амплитудной и частотной модуляции ЭМГ [6].

Процесс управления экзоскелетом начинается с сегментации текущего сигнала ЭМГ на не пересекающиеся окна шириной TW с последующим формированием из отсчетов каждого окна информативного признака FD . С этой целью осуществляют переход от текущего дискретного отсчета ЭМГ x_τ к отсчету y_τ , получаемому путем сравнения текущего отсчета с порогом Θ и вычисляемому согласно выражению

$$y_\tau = \begin{cases} 1, & \text{если } |x_\tau| \geq \Theta; \\ 0, & \text{если } |x_\tau| < \Theta. \end{cases} \quad (1)$$

На основе отсчетов (1) вычисляют информативный признак в окне TW как

$$FD = \frac{1}{TW} \sum_1^{TW} y_\tau. \quad (2)$$

На рис. 2 показано, как из сигнала ЭМГ $x(t)$ рис. 2а получается сигнал $y(t)$ рис. 2б для одного порога Θ . На ширине окна можно использовать множество порогов Θ и как следствие, получить множество информативных признаков FD .

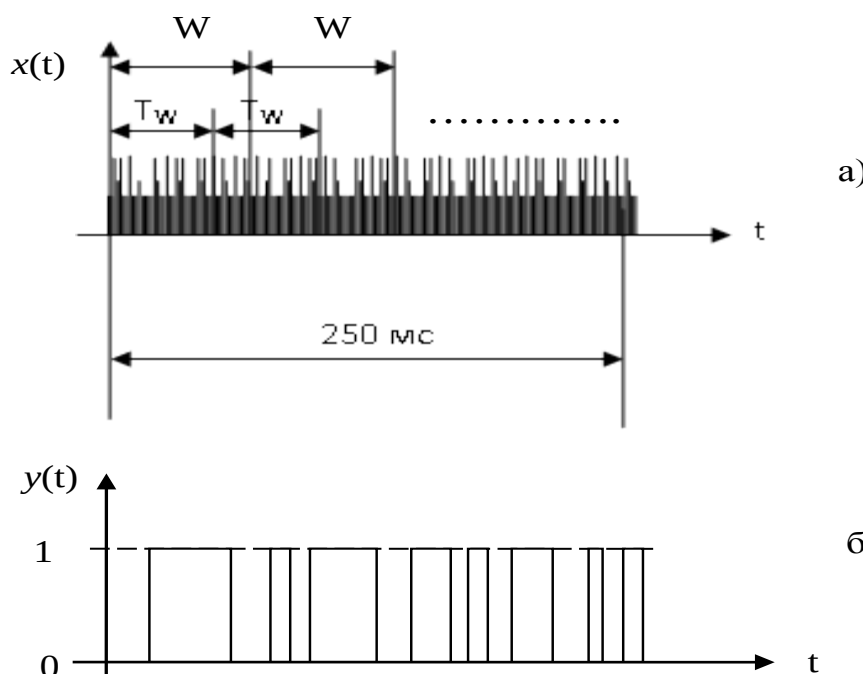


Рис. 2. Иллюстрация перехода от сигнала ЭМГ $x(t)$ (а) к сигналу $y(t)$ (б).

Таким образом, на апертуре окна TW формируется множество информативных признаков (2), на основе распознавания которых формируется соответствующая команда на серводвигатели 4 рисунок 1. Задавшись апертурой ЭМГ, на которой принимается решение о включении соответствующего серводвигателя, можем выделить на ней множество окон. Учитывая, что временная апертюра дешифрации ЭМГ сигнала составляет 250 мс, а минимальный информационный квант, то есть минимальный отрезок ЭМГ сигнала, который переносит релевантную информацию, составляет 25 мс, агрегатор решений работает на апертуре 250 мс. Так как решение, принимаемое по результатам анализа одного окна, является частным решением, то необходим агрегатор этих решений. Таким образом, приходим к выводу, что необходимы две ступени дешифрации ЭМГ.

Для осуществления дешифрации ЭМГ на двух ступенях используем обучаемые нейронные сети прямого распространения сигнала. Первая нейронная сеть ($NET1$) работает с пространством информативных признаков (2), сформированном в окне TW . В качестве агрегатора также используем обучаемую нейронную сеть прямого распространения, которая работает на апертуре $MW = \text{int}(250/W)$ или, если используются окна с перекрытием, на апертуре $MW = \text{int}((250 - TW)/W)$, где W – шаг перемещения окна на апертуре ЭМГ.

На рис. 3 представлен алгоритм управления экзоскелетом посредством анализа ЭМГ. Блоки 2-5 алгоритма осуществляют анализ окон ЭМГ на заданной апертуре сигнала. Результаты этого анализа накапливаются в блоке памяти 5. После анализа ЭМГ на заданной апертуре осуществляется переход на агрегатор $NET 2$, в котором принимается окончательное решение по команде на серводвигатели.

Так как выходы $NET1$ разнесены во времени, то для принятия решения по совокупности результатов этих выходов необходимо запоминающее устройство, в котором хранят эти результаты. Накапливание результатов классификации окон в $NET1$ осуществляется в блоке 5 рис. 3. После анализа соответствующего количества окон результаты этого анализа поступают на обучаемую нейронную сеть $NET2$ (блок 6), выходы которой подключены к контроллеру серводвигателей 4 (рис. 1). Число окон анализа, то есть входов $NET2$, зависит от времени дешифрации команды, зашифрованной в ЭМГ сигнале, и от минимального кванта информации, переносимого ЭМГ сигналом.

Результаты.

За основу реабилитационного процесса выбран процесс вертикализации пациента с опорой для рук, которая выступает в качестве простейшего реабилитационного изделия [1]. Для управления серводвигателями экзоскелета в процессе вертикализации используем пять каналов ЭМГ, которые считывают сигнал ЭМГ с большой ягодичной мышцы, двуглавой мышца бедра, полуперепончатой мышцы, полусухожильной мышцы, большой поясничной мышцы.

Процесс вертикализации управляется посредством дешифрации ЭМГ в каналах и скорость вертикализации зависит от интенсивности ЭМГ сигналов в каналах. При определенных сочетаниях интенсивностей ЭМГ в каналах серводвигатели могут остановиться в любой фазе вертикализации. Чувствуя, что у него «не хватает сил встать», пациент захочет сесть. В этом случае возможна генерация соответствующих сигналов ЭМГ, которые могут быть использованы для реверса серводвигателей. Для осуществления режима «сесть» используются каналы ЭМГ с подвздошно-поясничной мышцы.

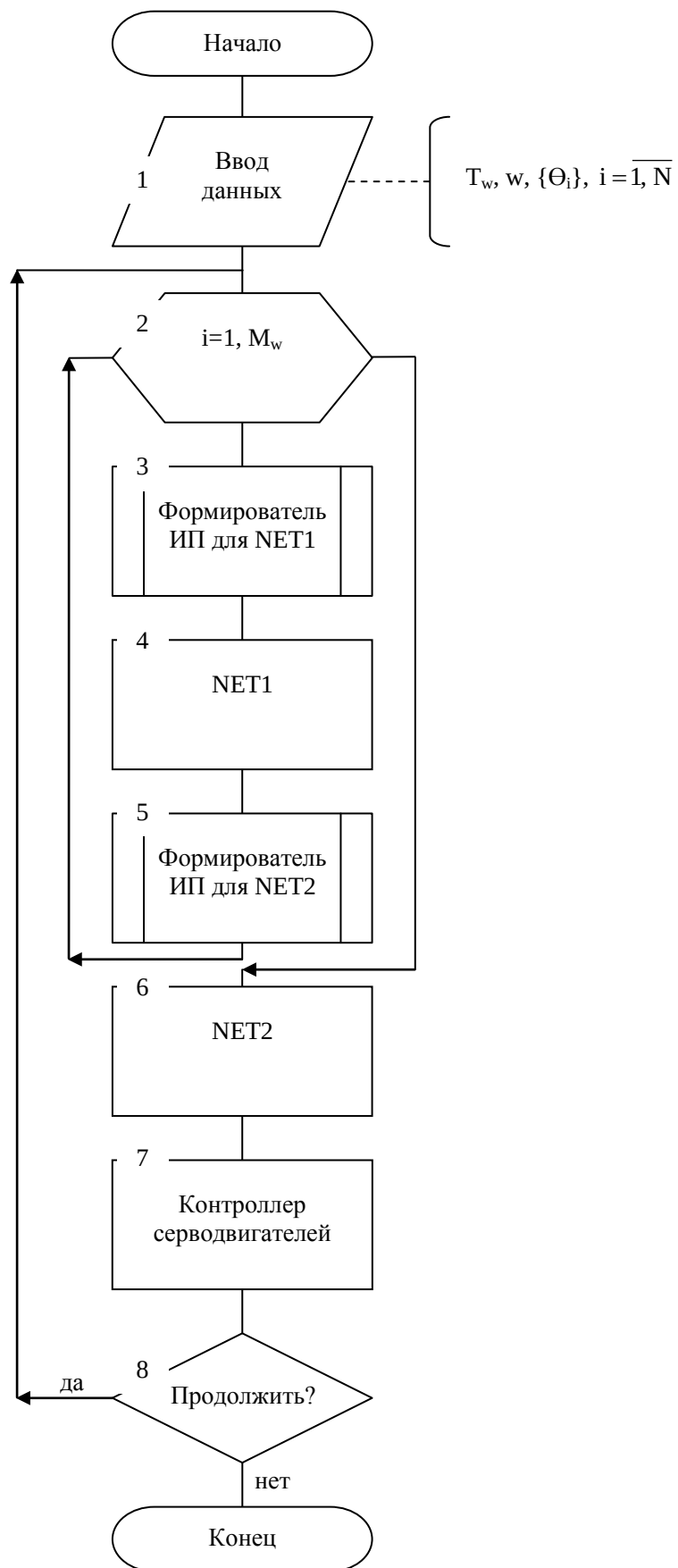


Рис. 3. Схема алгоритма управления экзоскелетом

Таким образом, для вертикализатора в качестве дешифратора ЭМГ сигналов необходимо использовать, как минимум, шесть каналов ЭМГ и еще одну нейросетевую структуру, используемую в качестве агрегатора дешифраторов каналов ЭМГ. Структурная схема такого дешифратора показана на рис. 5. Нейронная сеть *NET3* является нечеткой нейронной сетью. Ее выход вычисляется путем дефузификации шести функций принадлежности [9].

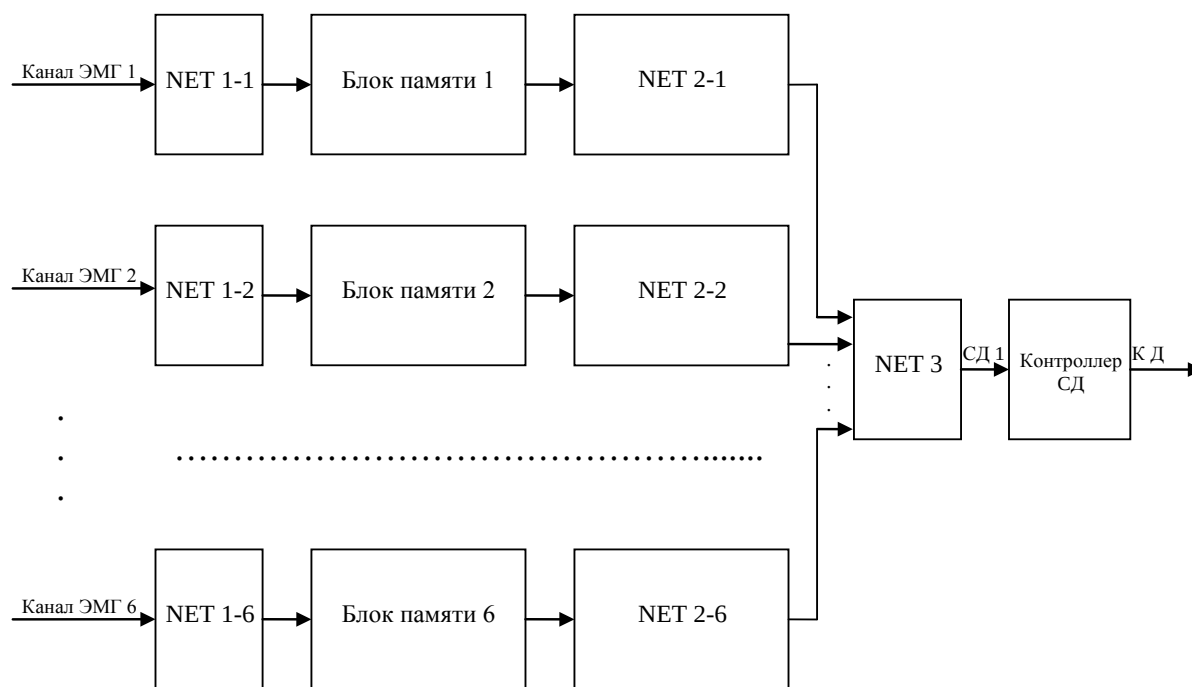


Рис. 4. Структурная схема классификатора ЭМГ при использовании дублирующих каналов

Управление вторым серводвигателем осуществляется контроллером серводвигателей на основе модели движения экзоскелета в режиме «встать – сесть». Параметры модели адаптируются под физическое состояние пациента, что позволяет устранить основной недостаток известных реабилитирующих экзоскелетов.

Выводы.

Для управления экзоскелетом в процессе реабилитации использовался многоканальный электромиосигнал, дешифрация которого в каждом канале осуществлялась посредством пошаговой сегментации на пересекающиеся или на не пересекающиеся окна. Дешифратор ЭМГ построен на основе двухуровневой нейросетевой структуры. Формирование вектора информативных признаков для нейронной сети первого уровня осуществлялось посредством многоуровневого компаратора, число уровней которого определяется размерностью вектора информативных признаков, а компоненты вектора информативных признаков определялись посредством процедуры клиппирования сигнала ЭМГ на различных уровнях компарации.

На основе базовой моделей классификатора осуществлено управление движением экзоскелета в режиме «встать – сесть». Определено расположение электродов на группе мышц для осуществления режима вертикализации с комбинированным управлением. Показано, что упрощенная кинематическая модель

режима вертикализации позволяет совместно с двухуровневой нейросетевой моделью дешифратора электромиосигналов осуществлять адаптацию реабилитационного процесса к функциональному состоянию пациента.

Полученные модели классификаторов поверхностных сигналов электромиограмм могут быть использованы при построении интеллектуальных систем реабилитации пациентов с неврологическими заболеваниями и позволят разрабатывать адаптирующиеся стимулирующие программы, тестирование результатов работы которых позволит разработать новые методы и средства для реабилитации пациентов с неврологическими заболеваниями.

Список литературы

1. Аль-Баред, А.Я.С. Алгоритмы синтеза оптимального управления в биотехнических системах реабилитационного типа на основе технологий нейронных сетей/ А.Я.С. Аль-Баред, А.Н. Брежнева, Р.А. Томакова // Системный анализ и управление в биомедицинских системах. – 2018. – Т. 17, №3. – С. 750 - 754.
2. Будко Р.Ю., Старченко И.Б. Создание классификатора мимических движений на основе анализа электромиограммы // Труды СПИИРАН. 2016. Вып. 46. С. 76-89.
3. Мобильная система управления экзоскелетом с помощью электромиографических сигналов мышц человека/ М.А. Хоружко, Г.Н. Сесекин, Н.В. Болдырева, М.О. Шамшин, И.А. Кастальский, В.И. Миронов, А.С. Пимашкин, В.Б. Казанцев// СТМ [2017 — том 9, №4. - С. 162-169.
4. Нейросетевые модули с виртуальными потоками для классификации и прогнозирования функционального состояния сложных систем / А.В. Киселев, Т.В. Петрова, С.В. Дегтярев, А.Ф. Рыбочкин, С.А. Филист, О.В. Шаталова, В.Н. Мишустин // Известия Юго-Западного государственного университета. 2018. № 4 (79). - С. 123-134.
5. Патент РФ RU 2 635 632 C1. Способ и система управления интеллектуальной бионической конечностью/Н.М. Иванюк, В.Р. Каримов, Р.Ю. Будко, П.В. Гронский, С.М. Клейман. Опубликовано 14.11.2017. Бюл. № 32.
6. Петрова, Т.В. Предикторы синхронности системных ритмов живых систем для классификаторов их функциональных состояний / Т.В. Петрова, С.А. Филист, С.В. Дегтярев, А.В. Киселев, О.В. Шаталова // Системный анализ и управление в биомедицинских системах. 2018. Т. 17. № 3. - С. 693-700.
7. Трифонов, А.А. Биотехническая система с виртуальной реальностью в реабилитационных комплексах с искусственными обратными связями/А.А. Трифонов, Е.В. Петрунина, С.А. Филист, А.А. Кузьмин, В.В. Жилин//Известия Юго-Западного государственного университета. Серия: Управление, вычислительная техника, информатика. Медицинское приборостроение. 2019; 9(4). – С.49-66.
8. Филист С.А. Кодовые образы сигналов электроэнцефалограммы для управления робототехническими устройствами посредством интерфейса мозг-компьютер/ С.А. Филист, Е.В. Петрунина, А.А. Трифонов, А.В. Серебровский//Моделирование, оптимизация и информационные технологии. Научный журнал, Том 7, № 1. - С.67-79.
9. Филист, С.А. Гибридная нейронная сеть с макрослоями для медицинских приложений / С.А. Филист, О.В. Шаталова, М.А. Ефремов // Нейрокомпьютеры. Разработка и применение. - 2014. - №6. - С. 35-39.

ОБЛАЧНАЯ СИСТЕМА ПОЛУАВТОМАТИЧЕСКОЙ РАЗМЕТКИ НАБОРОВ ДАННЫХ ДЛЯ ОБУЧЕНИЯ ГЛУБОКИХ НЕЙРОННЫХ СЕТЕЙ¹

И.С. Фомин¹, А.В.Бахшиев^{1,2}

¹ФГАНУ Центральный научно-исследовательский институт робототехники и технической кибернетики (ЦНИИ РТК), i.fomin@rtc.ru, alexab@rtc.ru

² Санкт-Петербургский политехнический университет Петра Великого (СПбПУ),
bakhshiev_av@spbstu.ru

Введение

При разработке интеллектуальных систем, основанных на нейронных сетях для решения задач в различных областях технического зрения остро встает вопрос о наборах данных для обучения нейронных сетей. Несмотря на существующие на сегодняшний день обширные наборы данных, всегда будут задачи, требующие создание специфических данных. Упрощение этой задачи позволит существенно расширить возможности исследователей в различных областях, в том числе мало знакомых с практикой применения нейронных сетей.

Далее будут рассмотрены существующие наборы данных и то как они создавались, а также существующие алгоритмы классификации и обнаружения объектов. Также будет приведена архитектура предлагаемой облачной системы полуавтоматической разметки.

Методы формирования наборов данных

Набор данных MNIST [1], содержит бинарные изображения рукописных цифр, общим объемом в 60000 примеров. Всего в наборе 500 примеров различных почерков, 250 собраны в обучающем и 250 в тестовом. Исходные цифры размером 20x20 пикселей центрированы в изображениях размером 28x28 пикселей, конечные изображения приведены в оттенках серого. Набор Fashion-MNIST [2] позиционирует себя как замена набору MNIST, позволяющая проводить тесты на наборе по сложности соответствующем современности, а по интерфейсу не отличающемся от MNIST. С использованием данных сайта [3] собраны изображения предметов гардероба, которые были конвертированы в формат PNG, фоновые области были обрезаны, размеры изменены так чтобы по длинной стороне было 28 пикселей, повышена контрастность, размер был расширен до 28x28, применена инверсия и конверсия в 8-битное серое изображение. Класс предмета определяется силуэтом, всего в наборе 10 классов: футболка, брюки, пуловер, платье, куртка, сандалии, рубашка, кроссовки, сумка, полуботинки.

Для соревнования PASCAL VOC Challenge был собран набор [4], включающий помимо прочих набор для обнаружения. В наборе присутствуют аннотации для 20 классов. Попытка использования краудсорсинга не дала достаточное качество, была применена только для сбора информации по классам, разметка прямоугольников осуществлялась самими исследователями каждый год в группах под руководством опытных аннотаторов.

Набор ImageNet [5] на момент появления являлся самым крупным и амбициозным проектом по сбору данных. Состав классов основан на тезаурусе английского языка WordNet [6]. Первичный сбор информации осуществлялся автоматически с Flickr, до

¹ Работа проводилась в рамках выполнения государственного задания Минобрнауки России № 075-01195-20-00 “Облачные сервисы автоматического синтеза и валидации наборов данных для обучения глубоких нейронных сетей в задачах распознавания образов”

10000 изображений на класс. Разметка наличия классов проводилась через Amazon Mechanical Turk (AMT), каждое изображение несколько раз демонстрировалось разным пользователям, правильный класс определялся статистически на основе специально разработанного алгоритма для анализа достаточного количества положительных решений. В наборе для локализации объектов к классам добавлены ограничивающие прямоугольники. Контроль качества осуществлялся путем верификации каждого прямоугольника. Задача разметки изображения делилась на подзадачи рисования (выделить объект), верификации качества (хорошая ли разметка) и верификации покрытия (все ли экземпляры отмечены). Это позволило получить прозрачную процедуру разметки с понятными затратами времени.

Для задачи обнаружения в ILSVRC [7] было отобрано 200 классов объектов, каждый представляющий достаточно простую категорию. Главная сложность здесь – отметить все классы на всех изображениях набора. Подход с просьбами пользователям найти и отметить каждый из классов приведет к огромному числу (по 200 на изображение) запросов, что неприемлемо. Использованы подходы с использованием корреляции (некоторые классы часто попадаются вместе), иерархии (некоторые классы могут быть сгруппированы, и размечать можно наличие группы) и разреженности (небольшое число классов на изображение, можно отмечать отсутствие отдельных классов и групп классов). Были разработаны стратегии формирования последовательностей запросов чтобы за минимальное число шагов определить все классы на изображении. Запросы собраны так, чтобы ошибочно верные утверждения влияли только на количество проверок одного изображения. Когда все классы для всех изображений были отобраны, разметка осуществлялась так же как и при локализации.

Набор OpenImages [8] содержит 9.2 млн изображений, содержащих данные классификации и обнаружения. Данные набора собраны с Flickr, удалены те, которые легко находятся поисковыми движками для усложнения набора, сильно схожие между собой. Набор классов унаследован от внутренней сети Google для классификации JFT (всего 19794 класса). Для обнаружения были отобраны 600 классов, которые можно физически выделить ограничивающим прямоугольником. Ручная разметка 19 тысяч классов для классификации физически неосуществима, невозможно запомнить отличия всех. К изображениям валидационного и тестового наборов применялся классификатор, обученный на JFT, пользователи AMT верифицировали разметку вручную. В наборе для обучения к каждому изображению применялись десятки классификаторов, обученных на JFT, после чего проводилась стратифицированная верификация операторами части отметок. Для разметки данных обнаружения были сделаны гайды что считать классом, размечены примеры правильного выделения и примеры частых ошибок, один оператор за один присест работал только с одним классом. Прямоугольники выделялись не рамкой, а крайними левой/правой/верхней/нижней точками, что существенно ускорило процесс. Краудсорсинг не использовался. Операторов обучали использовать выделение точками на наборе PASCAL VOC. Часть данных размечалась с использованием нейросетевых детекторов и верификации с помощью оператора.

Набор Tencent ML-Images [9] составлен частично из изображений ImageNet, частично из OpenImages. В объединенном наборе присутствует 11166 классов, 17,61 млн изображений в обучающем, 88,74 тысячи в тестовом наборах. После объединения выбранные метки классов были распространены на весь набор.

Также могут применяться средства формирования синтетических данных, например [10], но они более полезны для предварительного обучения классификации редких категорий для последующего использования для помощи оператору и обучения в процессе первых сессий аннотирования.

Из приведенных выше примеров аннотирования можно сделать выводы о том, что даже в самых объемных наборах в цикле разметки присутствует оператор, и это следует учитывать при формировании системы формирования разметки. Даже обучение на наиболее внутреннем наборе Google JFT не гарантирует максимального качества, и частичная верификация неизбежна. При разработке важно создание интеллектуальных алгоритмов, которые смогут минимизировать участие человека в процессе без потери качества, в идеале – исключить действия оператора из цикла аннотирования для простых классов, в том числе за счет обучения групп классификаторов присутствия/отсутствия класса и их комплексирования.

Методы классификации и обнаружения

На сегодняшний день разработано большое разнообразие архитектур нейронных сетей для классификации объектов. Самые первые известные разработки AlexNet [11] и VGG-Net [12] появились в 2012 и 2013 годах соответственно и с тех пор развитие области только набирало обороты. Проблема насыщения роста качества с ростом глубины сети была решена с помощью модуля Inception в сети GoogLeNet [13], позволившем данным проходит разными путями внутри сети, и количество слоев со временем достигло сотен, в работах Inception V2 [14] и Inception V3 [15], в которых модули Inception были доработаны для повышения производительности. Похожим способом развивалась архитектура ResNet [16], где к группам слоев добавлялись обходные соединения. Развитие сетей путем роста количества слоев привело к резкому снижению производительности и росту сложности вычислений, что привело к созданию облегченных нейронных сетей, таких как SqueezeNet [17] и MobileNets [18]. Разработки выполняются до сих пор, некоторые разработчики (NASNet [19]) пытаются использовать алгоритмы для генерации структуры сети при решении простых задач. Другие авторы (DenseNet [20]) наращивают плотность связей для передачи признаков с нижних слоев к верхним, что позволяет увеличить точность.

Методы обнаружения объектов на изображениях

Через некоторое время после начала активного развития сетей для классификации объектов появляются попытки применить некоторые решения для обнаружения объектов. После появления решений для выделения областей на изображении следующим шагом стало появление R-CNN [21], выделяющей предположения регионов и потом классифицирующей их, затем Fast R-CNN [22], где изображение обрабатывалось сверточными слоями только один раз, а не для каждого из предположений отдельно. Замедление работы за счет использования внешнего средства для предположения регионов было решено путем создания сети предположения регионов RPN и привело к созданию Faster R-CNN [23], полностью нейронной архитектуры.

Идея с многократным использованием выходов сверточных слоев нашла свое применение в архитектурах one-shot детекторов, когда для выходов сверточного слоя генерируется большой массив прямоугольников, для каждого считается принадлежность классу, точность и регрессия ограничивающего прямоугольника, а потом выбираются наиболее релевантные выходы. Так работали первые варианты SSD [24] и YOLO [25]. Доработанная версия YOLOv2 [26] исключила из состава полносвязные слои еще более повысив качество работы и позволила сделать вариант архитектуры с распознаванием 9000 классов. Также был представлен вариант YOLOv3 [27], вобравший в себя идеи RetinaNet [28] для дальнейшего увеличения качества.

Архитектуры обнаружения также стремились сделать маленькими и эффективными, это привело к появлению детекторов SqueezeDet [29], сочетавшего идеи YOLO и SqueezeNet, показавшего высокое качество при малом размере и большом быстродействии, и сетей MobileNets в рамках фреймворков обнаружения в том числе.

Интеллектуальная система формирования наборов данных

Прогресс нейросетевых технологий в области классификации изображений и обнаружения объектов на изображениях достиг столь впечатляющих высот, что некоторые эксперты считают эти задачи принципиально решенными. Рассмотренные выше алгоритмы для классификации и обнаружения вместе с существующими наборами данных позволяют решать широкий класс задач. Обученные сети применимы том числе для формирования аннотаций на новых изображениях с заданным числом классов.

Рассмотрим основные требования, предъявляемые к системе формирования аннотаций, а также пути их решения.

1. Такая система должна быть доступна из многих мест, так как даже в рамках небольшого проекта с ней будет работать некоторое количество специалистов. Это означает наличие сервера с доступом к нему через интернет, а также клиентского ПО или иных способов взаимодействия.

2. Наличие возможности выполнять несколько расчетов одновременно. Для разрешения данного требования есть несколько способов, наиболее разумным из которых представляется распределение вычислений на несколько ПК, расчеты на которых выполняются независимо.

3. Хранение большого массива данных, включающего в себя размеченные наборы данных, данные, загруженные пользователями для обработки, результаты обработки и предобученные веса. Для хранения данных в системе используется распределение между несколькими крупными хранилищами, а также частично задействуются вычислители.

4. Централизованное управление распределенной системой. Так как и данные и вычисления распределены в несколько мест требуется единый центр управления и хранения информации о системе. В качестве такого центра используется центральный сервер, содержащий распределенную СУБД, которая хранит информацию обо всех зарегистрированных в системе вычислителях.

Интерфейс пользователя позволяет контролировать серверную часть при наличии прав доступа, контролировать работу системы, размечать новые данные и исправлять результаты разметки обоих типов.

Качество работы обученного нейросетевого алгоритма зависит от архитектуры, гиперпараметров сети, и, главное, репрезентативности обучающего набора относительно поступающих на вход данных. Подготовка обучающего набора состоит из нескольких этапов. На этапе сбора и фильтрации начальных данных набора нейронные сети используются для оценки их разнообразия и репрезентативности. В случае классификации если существуют обученные классификаторы нужных классов строится набор с малым количеством ошибок с использованием описанных выше алгоритмов валидации. Для наборов обнаружения объектов при наличии обученных детекторов нужных классов можно провести предварительную разметку с последующей автоматической и ручной валидацией.

Для автоматизации и упрощения формирования разметки накоплен широкий спектр частных и общих нейросетевых решений с наборами весов для обнаружения и классификации как общими, для работы с большим числом классов, так и частными, для высокоточного распознавания малого числа классов. С ростом числа данных в системе растет количество обученных весов и возможности системы. Когда решение будет опубликовано его возможности будут расти с ростом числа пользователей.

Заключение

Предложенная система решает сразу несколько проблем, связанных с формированием наборов данных. Полуавтоматическая разметка данных сократит финансовые и временные затраты на разметку наборов данных.

В систему заложены готовые наборы данных, архитектуры и веса рассчитанные на обнаружение и классификацию большого количества категорий объектов. С ростом количества размеченных пользователями наборов данных вырастет количество классов, с которыми может работать система и общее качество обнаружения и классификации за счет постоянного дообучения и переобучения на новых данных.

Список литературы

1. LeCun Y. Gradient-based learning applied to document recognition / LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. // Proceedings of the IEEE. – 1998. – Т. 86. – №. 11. – С. 2278-2324.
2. Xiao H., Rasul K., Vollgraf R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. E-print: <https://arxiv.org/pdf/1708.07747.pdf>,
3. Zalando: Shoes and Fasion Online, <https://zalando.com/>.
4. Everingham M. The pascal visual object classes challenge: A retrospective. / Everingham M, Eslami S. A., Van Gool L., Williams C. K., Winn J., Zisserman A. // International journal of computer vision. – 2015. – Т. 111. – №. 1. – С. 98-136.
5. Deng J., Imagenet: A large-scale hierarchical image database. / Deng J., Dong W., Socher R., Li L. J., Li K., Fei-Fei L. // 2009 IEEE conference on computer vision and pattern recognition. – Ieee, 2009. – С. 248-255.
6. WordNet: An electronic lexical database, <https://wordnet.princeton.edu/>.
7. Russakovsky O. Imagenet large scale visual recognition challenge / Russakovsky O., Deng J., Su H., Krause J., Satheesh S., Ma S.,...,Berg A. C. // International journal of computer vision. – 2015. – Т. 115. – №. 3. – С. 211-252.
8. Kuznetsova A., Rom H., Alldrin N., Uijlings J., Krasin I., Pont-Tuset J.,..., Ferrari V.: The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. E-print: <https://arxiv.org/pdf/1811.00982.pdf>.
9. Wu B. Tencent ml-images: A large-scale multi-label image database for visual representation learning / Wu B., Chen W., Fan Y., Zhang Y., Hou J., Liu J., Zhang T. //IEEE Access. – 2019. – Т. 7. – С. 172683-172693.
10. Ros G. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes / Ros G., Sellart L., Materzynska J., Vazquez D., Lopez A. M. //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – С. 3234-3243.
11. Krizhevsky A. Imagenet classification with deep convolutional neural networks / Krizhevsky A., Sutskever I., Hinton G. E. //Advances in neural information processing systems. – 2012. – С. 1097-1105.
12. Simonyan, K., Zisserman, A. Very deep convolutional networks for large-scale image recognition. E-print: <https://arxiv.org/pdf/1409.1556.pdf>
13. Szegedy C. Going deeper with convolutions / Szegedy C., Liu W., Jia Y., Sermanet P., Reed S., Anguelov D.,..., Rabinovich A. //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – С. 1-9.
14. Ioffe, S., Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. E-print: <https://arxiv.org/pdf/1502.03167.pdf>

15. Szegedy C. Rethinking the inception architecture for computer vision. / Szegedy C., Vanhoucke V., Ioffe S., Shlen J., Wojna Z. //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – C. 2818-2826.
16. He K. Deep residual learning for image recognition / He K., Zhang X., Ren S., Sun J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, (2016).
17. Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., Keutzer, K.: SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size. E-print: <https://arxiv.org/pdf/1602.07360.pdf>
18. Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T.,..., Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. E-print: <https://arxiv.org/pdf/1704.04861.pdf>
19. Zoph B. Learning transferable architectures for scalable image recognition / Zoph B., Vasudevan V., Shlens J., Le Q. V. //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2018. – C. 8697-8710.
20. Huang G. Densely connected convolutional networks / Huang G., Liu Z., Van Der Maaten L, Weinberger K. Q. Huang G. // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2017. – C. 4700-4708.
21. Girshick R. Rich feature hierarchies for accurate object detection and semantic segmentation / Girshick R., Donahue J., Darrell T., Malik J. //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2014. – C. 580-587.
22. Girshick R. Fast r-cnn / Girshick R. //Proceedings of the IEEE international conference on computer vision. – 2015. – C. 1440-1448.
23. Ren S. Faster r-cnn: Towards real-time object detection with region proposal networks / Ren S., He K., Girshick R., Sun J. //Advances in neural information processing systems. – 2015. – C. 91-99.
24. Liu W. Ssd: Single shot multibox detector Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu C. Y., Berg A. C. //European conference on computer vision. – Springer, Cham, 2016. – C. 21-37.
25. Redmon J. You only look once: Unified, real-time object detection / Redmon J., Divvala S., Girshick R., Farhadi A. // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – C. 779-788.
26. Redmon J. YOLO9000: better, faster, stronger / Redmon J., Farhadi A. //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2017. – C. 7263-7271.
27. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. E-print: <https://arxiv.org/pdf/1804.02767.pdf>
28. Lin T. Y. Focal loss for dense object detection / Lin T. Y., Goyal P., Girshick R., He K., Dollár P. //Proceedings of the IEEE international conference on computer vision. – 2017. – C. 2980-2988.
29. Wu B. Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving Wu B., Iandola F., Jin P. H., Keutzer K. //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. – 2017. – C. 129-137.

НЕЙРОСЕТЕВОЙ МЕТОД АДАПТАЦИИ КОНТЕНТА В ЗАВИСИМОСТИ ОТ СОСТОЯНИЯ ПОЛЬЗОВАТЕЛЕЙ

Л.В.Хливненко¹, Ф.А.Пятакович²

¹Воронежский государственный технический университет,
Факультет машиностроения и аэрокосмической техники, каф.прикладной математики и механики, hlivnenko_lv@mail.ru

²Белгородский государственный национальный исследовательский университет,
Медицинский институт, каф.пропедевтики внутренних болезней и клинических информационных технологий, piatakovich@gmail.com

В настоящее время наблюдается всплеск интереса к дистанционным технологиям обучения. Качественно организованная информационно-образовательная среда позволяет получать новые знания с минимальными временными и финансовыми затратами. Автоматизированный процесс получения знаний предполагает обновленный формат взаимодействия обучающихся и обучающихся.

Одной из основных проблем такой формы взаимодействия является удержание внимания к процессу обучения. Обучающийся должен обладать высоким уровнем самодисциплины и способности к концентрации внимания.

Другой существенной проблемой является организация навигации в информационном потоке. В этой связи обучающемуся пользователю необходима помощь тьютора в выстраивании индивидуальных образовательных траекторий. Эту функцию может взять на себя интеллектуальный агент, основу которого составляет когнитивная архитектура.

Такой виртуальный агент в системах дистанционного обучения может выступать в качестве персонального менеджера обучающегося, адаптируя образовательный контент в зависимости от психофизиологического состояния ученика.

При этом важным моментом является косвенный сбор данных о психофизиологическом состоянии.

Умный «головной» обруч, разработанный компанией BrainCo Inc. и проходящий тестирование в Китае для контроля внимательности учеников, вряд ли будет способствовать повышению комфортности процесса обучения. Девайсы, как устройства, работающие на основе анализа ЭЭГ [1], пока не очень удобны на практике, поскольку располагаются на голове. Сканирование мозговой активности через волосы часто мешает качественному считыванию сигнала.

К косвенным данным об утомляемости обучающегося можно отнести айтрекинг, клавиатурный почерк, вариабельность сердечного ритма (ВСР).

Анализ ВСР является неинвазивным и удобным для пользователя методом взаимодействия с обучающей системой. Ученик охотнее наденет на руку модный фитнес-браслет чем «шапочку» из электродов.

Целью данного исследования является разработка рекомендаций по адаптации образовательного контента на основе использования нейросетевого метода распознавания степени активности автономной нервной системы человека.

Автономная (вегетативная) нервная система (АНС) контролирует все жизненно важные функции организма без непосредственного участия сознания (воли) человека. АНС состоит из двух отделов: симпатического и парасимпатического.

Симпатическая нервная система (СНС) отвечает за мобилизацию ресурсов организма и активизируется в стрессовых ситуациях. Парасимпатическая нервная система (ПСНС), напротив, отвечает за расслабление и активизируется в случае покоя.

Известны шесть степеней активности АНС человека: резко выраженное преобладание СНС, выраженное преобладание СНС, умеренное преобладание СНС, равновесное состояние между симпатическим и парасимпатическим отделом АНС, умеренное преобладание ПСНС, выраженное преобладание ПСНС.

Исходными данными для принятия решений служат вектора кардиоинтервалов RR_i , $i = \overline{1,500}$, полученные с помощью датчика пульса. RR_i - это время в секундах между высокоамплитудными R-зубцами ЭКГ. Текущая активность СНС и ПСНС напрямую связана с реакцией многоконтурной и многоуровневой системы регуляции кровообращения [2].

Разность $(RR_{i+1} - RR_i)$ называется «коррекцией». Все наблюдаемые значения коррекций у человека попадают в интервал $(-1,55; 1,55)$. Вариационный ряд разбивается на равные участки длиной 0,05 сек. Получается 61 класс дифференциальной гистограммы распределения микроструктуры паттерна variability ритма сердца, включающий норму, тахи- и брадиаритмию.

Входами модели x_j , $j = \overline{1,61}$, служат частоты полученного интервального вариационного ряда. Координаты входного вектора $X = \{x_j\}$ подвергаются процедуре «нормализации» вычитанием их выборочного среднего и нормированием на исправленное выборочное среднее квадратичное отклонение.

Мониторинг состояния АНС осуществляет двухслойная искусственная нейронная сеть (ИНС) прямого распространения, состоящая из 61 входа (x_j), 10 нейронов скрытого слоя (y_k^c , $k = \overline{1,10}$) и 6 нейронов выходного слоя (y_i , $i = \overline{1,6}$).

Функционирование ИНС описывается формулами (1) – (2).

$$y_k^c = f\left(\sum_{j=1}^{61} w_{kj} x_j\right), \quad (1)$$

где w_{kj} - весовой коэффициент связи между j -м входом и k -м нейроном скрытого слоя. В качестве активационной функции здесь и далее выбран гиперболический тангенс: $f(s) = (e^s - e^{-s}) / (e^s + e^{-s})$.

$$y_i = f\left(\sum_{k=1}^{10} v_{ik} y_k^c\right), \quad (2)$$

где v_{ik} - весовой коэффициент связи между k -м нейроном скрытого слоя и i -м нейроном выходного слоя.

Максимальное вычисленное значение y_i определяет номер класса i , к которому ИНС относит входной паттерн кардиоинтервалов. Номер класса соответствует одному из шести состояний АНС.

Наборы весовых коэффициентов $\{w_{kj}\}$, $\{v_{ik}\}$, используемые в режиме диагностики, получаются в результате предварительного обучения ИНС на обучающем множестве, расклассифицированном экспертом.

В исследовательских целях в качестве обучающего множества использовалась пробная выборка, состоящая из 168 записей кардиоинтервалов людей, входящих в разные социальные и возрастные группы [3].

В целях обучения ИНС врач-эксперт, анализируя форму скаттерграммы, построенной по паттернам кардиоинтервалов из обучающего множества, распределил все примеры по 6 целевым классам. В целевых выходных векторах позиция t_i «верного» класса маркировалась значением 0,5. Остальные координаты принимали значения: -0,5.

Обучение ИНС проводилось путем комбинирования градиентного и стохастического методов обучения.

Корректировка весовых коэффициентов выполнялась по формулам (3)-(4).

$$v_{ik}^{N+1} = v_{ik}^N - \eta \alpha (\delta_i)_l^N (y_k)_l^N + (1-\eta) v_c, \quad (3)$$

$$w_{kj}^{N+1} = w_{kj}^N - \eta \alpha \left(\sum_{i=1}^6 (\delta_i)_l^N v_{ik}^N \right) \left(1 - ((y_k)_l^N)^2 \right) (x_j)_l^N + (1-\eta) w_c, \quad (4)$$

где N - дискретный момент времени, η - коэффициент, управляющий относительными величинами градиентной и стохастической компонент веса, α - скорость обучения, $\delta_i = (y_i - t_i)(1 - y_i^2)$, v_c и w_c - стохастические коррекции соответствующих весов, определяемые формулой (5).

$$v_c = w_c = \alpha T(N) \operatorname{tg}(P(w)), \quad (5)$$

где $T(N)$ - искусственная температура, рассматриваемая как функция времени:

$$T(N) = \frac{T_0}{1 + N}, \quad (6)$$

$P(w)$ - это случайное число из равномерного распределения на интервале $(-\pi/2; \pi/2)$. Отрицательная часть введена в рассмотрение для случайного определения знака коррекции.

После корректировки весовых коэффициентов осуществлялся перерасчет значения целевой функции, определяемой формулой (7).

$$E(v(N), w(N)) = \left(\sum_{i=1}^6 \left(f \left(\sum_{k=1}^{10} v_{ik}(N) f \left(\sum_{j=1}^{61} w_{kj}(N) x_j^{(N)} \right) \right) - t_i^{(N)} \right)^2 \right) / 2. \quad (7)$$

Если ошибка сети уменьшалась, то такие изменения весовых коэффициентов сохранялись. Чтобы процесс обучения сети не застревал в локальных минимумах целевой функции, а двигался в направлении поиска глобального минимума, при увеличении ошибки сети новые весовые коэффициенты сохранялись с вероятностью, определяемой распределением Коши:

$$p(w) = \frac{T(N)}{T^2(N) + w^2}, \quad (8)$$

где w - стохастическая коррекция веса.

Интегрируя $p(w)$ в равенстве (8) от 0 до w и разрешая полученное уравнение относительно w , имеем формулу (5), где $P(w) = \int_0^w p(w) dw$.

При инициализации сети все весовые коэффициенты принимали случайные значения из отрезка $[-0,3; 0,3]$. Скорость обучения $\alpha = 0,1$.

Апробация представленного метода распознавания степени активности АНС была проведена в свободно распространяемой среде программирования Lazarus. Авторами было разработано полнофункциональное приложение, позволяющее формировать случайные подвыборки из имеющегося набора данных, обучать нейронную сеть, тестировать эффективность обучения на контрольных подмножествах [4].

В результате многократного обучения сети был найден набор значений весовых коэффициентов, для которого выполняется 100% верное распознавание по всей пробной выборке.

Результаты и их обсуждение.

Представленный нейросетевой метод распознавания активности АНС на основе анализа ВСР позволяет осуществлять мониторинг функционального состояния человека в режиме реального времени в достаточно комфортном для него формате.

Персонализированная обучающая система (виртуальный агент), «наблюдая» изменения активности АНС, варьирует подачу образовательного контента.

В состоянии покоя у более молодых людей наблюдается равновесное состояние между симпатическим и парасимпатическим отделом АНС. Умеренное преобладание СНС и ПСНС характеризует индивидуальные особенности организма. Важным моментом является дрейф функционального состояния в процессе обучающей нагрузки, которую можно рассматривать как стрессовый фактор.

Выделяют три стадии стресса [5]. На первой стадии тревоги или стадии экстренной мобилизации наблюдается активизация СНС. Вторая стадия резистентности или стадия устойчивого приспособления предполагает умеренные колебания между активизацией СНС и ПСНС. Третья стадия истощения возникает как ответная реакция на сверхсильные или сверхдлительные нагрузки и сопровождается крайними состояниями: выраженным преобладанием СНС в случае эмоционального раздражения или выраженным преобладанием ПСНС в состоянии эмоционального ступора.

Понимание физиологических процессов позволяет заблаговременно до наступления усталости и неприятия процесса обучения «разрядить» обстановку, сменив тактику обучения.

В случае старта обучающего сеанса с преобладания активности СНС обучающая система предлагает релаксирующие модули (игру, легкие задания на повторение, общение) до стабилизации равновесного состояния АНС или умеренного преобладания одной из компонент. Именно в этих границах реализуется подача основного обучающего контента.

Если изначально ученик сильно расслаблен с выраженным преобладанием активности ПСНС ему предлагаются тонизирующие модули (игра, тест, наставник) до стабилизации функционального состояния. Если этого не происходит, то обучающий сеанс переносится на другое время. Таким же образом обучающая система поступает при приближении функционального состояния к критическому (выраженному преобладанию активности СНС или ПСНС).

Список литературы

1. Shishkin S.L. Combining the eye gaze, EEG and MEG for creating new modes of humancomputer interaction / S.L. Shishkin. — The 5th International conference BCI: Science and Practice, Samara, RF, 2019. — P.3-4.
2. Баевский Р.М. Анализ variability сердечного ритма при использовании различных электрокардиографических систем / Р.М.Баевский, Г.Г.Иванов, Л.В.Чирейкин и др. // Вестник аритмологии. — 2001. — № 24. С.65 – 87.
3. Хливненко Л.В. Решение задачи классификации комбинированием градиентного и стохастического методов обучения искусственной нейронной сети / Л.В. Хливненко, Ф.А. Пятакович, В.В. Васильев, Т.И. Якунченко // Системный анализ и управление в биомедицинских системах. — 2015. — Т. 14. — № 3. С. 544-548.
4. Пятакович Ф.А. Структура нейросетевого модуля для исследования математической модели автоматической классификации функциональных состояний человека / Ф.А. Пятакович, Л.В. Хливненко, Т.И. Якунченко, В.В. Васильев // Современные проблемы науки и образования. — 2015. — № 2. Электронный ресурс. URL: www.science-education.ru/129-21845 (дата обращения: 18.11.2015).
5. Лапкин М.М. Избранные лекции по нормальной физиологии / М.М.Лапкин, Е.А.Трутнева. — М.: ГЕОТАР-Медиа. — 2018. — 237 с.

ПРОГНОЗ РИСКА СЕРДЕЧНО-СОСУДИСТЫХ ОСЛОЖНЕНИЙ С СЕГМЕНТИРОВАННЫМ ПРОСТРАНСТВОМ ФАКТОРОВ РИСКА И СИНЕРГЕТИЧЕСКИМИ КАНАЛАМИ¹

О.В. Шаталова, Д.А. Медников, З.У. Протасова, Н.С. Стадниченко
Юго-Западный государственный университет,
ФФиПИ, каф. биомедицинской инженерии, *Shatolg@mail.ru*

Введение

Диагностика сердечно-сосудистого риска (ССР) принадлежит к числу важнейших задач кардиологии [1]. Современные системы стратификации ССР акцентированы на традиционные факторы риска (ФР) и методы математического моделирования. Это не позволяет им обеспечивать надежное предсказание возникновения заболеваний. Поэтому перспективным является включение ряда новых ФР в прогностические модели, которые могут быть использованы в системах стратификации риска у лиц, не имеющих клинических проявлений сердечно-сосудистых заболеваний (ССЗ), а также разработки новой методологии построения математических моделей ССР.

1. Методы

За основу построения интеллектуальной системы поддержки принятия решений по прогнозированию ССР взяты мультиагентные информационные системы (МИС) [2-4]. В эти системы входят множество автономных интеллектуальных агентов (АИА). АИА, находящиеся на более высоких иерархических уровнях, интегрируют информацию структурно связанных с ними АИА нижних иерархических уровней. Веса АИА нижнего иерархического уровня определяются путем настройки АИА верхнего иерархического уровня. В качестве АИА могут использоваться обучаемые нейронные сети или системы нечеткого логического вывода [5, 6].

Однако в ряде исследований показано, что ряд ФР, вследствие синергетического эффекта, могут значительно усилить риск прогнозируемого заболевания. Поэтому модели МИС должны предусматривать учет такого взаимодействия.

Для построения базовой структуры МИС будем использовать АИА нижнего уровня, способные формировать предикторы ССР. В качестве базовой модели предикторов риска ССР взята диагностическая модель, представленная в работах [7]. Согласно этой модели ФР ССР разбиваются на четыре группы. В первую группу входят традиционные ФР (физикальные исследования, опрос), во вторую – группа инструментальных исследований (ЭКГ), в третью группу – ФР, обусловленные сопутствующими заболеваниями, и в четвертую – факторы окружающей среды (факторы профессионального риска) [6-8].

При прогнозировании ССР сопутствующие заболевания или синдромы могут быть уже установленным фактом, или нуждаться в диагностике. Отсюда следует, что МИС должна содержать элементы как диагностических, так и прогностических моделей. При этом необходимо отметить, что ФР диагностируемых заболеваний или синдромов являются также предикторами прогнозируемого заболевания. Таким образом, в зависимости от того, поставлен ли диагноз сопутствующего заболевания, структура МИС должна перестраиваться.

В качестве АИА нижнего иерархического уровня используются «слабые» классификаторы в виде одноместных функций принадлежности. Однако существует ряд

¹ Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 19-38-90116.

ФР – сопутствующих заболеваний, которые имеют сложную структуру, так как зависят от совокупности других ФР. Поэтому «слабый» классификатор в этом случае не является одностойной функцией принадлежности и для его получения необходим агрегатор. Функции принадлежности по выделенным ФР синтезировались высококвалифицированными экспертами. Экспертная группа выбиралась в соответствии с рекомендациями работы [9]. Агрегатор «слабых» классификаторов, построенных на основе одностойных функций принадлежности, формировался по схеме Е. Шортлиффа [8].

2. Результаты

Апробация МИС осуществлялась на оценке риска ишемической болезни сердца (ИБС). Исследовались пациенты, проходившие обследование или находящиеся на стационарном лечении в Курской областной клинической больнице №1 и в клинической больнице РЖД-Медицина города Курск. В экспериментальные исследования были включены 380 мужчин и женщин в возрасте от 21 до 60 лет. Обучающие и контрольные выборки формировались на основе всестороннего обследования пациентов высококвалифицированными врачами-специалистами. По результатам оценки ишемического риска (ИР) было установлено, что среди пациентов экспериментальной группы у 37% установлен высокий ИР, средний риск установлен у 33% пациентов из экспериментальной группы, низкий риск — у 30% пациентов из экспериментальной группы. Из полученной экспериментальной группы были сформированы обучающие и контрольные выборки для получения показателей качества принятия решений сформированных с применением предлагаемых методов и моделей и разработанной на их основе компьютерной программе решающих модулей.

При выборе состава ФР прогнозирования ИР исходим из того, что ишемические поражения чаще всего носят системный характер, при котором развитие периферических ишемических осложнений несёт за собой ишемию центральной гемодинамической системы. При этом происходит нелинейное взаимовлияние систем друг на друга, приводящее к взаимоотягощению ишемических процессов, вплоть до летального исхода [7]. Поэтому важен учет синергетического эффекта ФР ИР. В связи с этим в качестве базовой системы прогнозирования ИР, была взята диагностическая модель, представленная рис. 2.

Эта модель учитывает, что факторами ИР могут быть не только определенные биомедицинские показатели, но и уже имеющиеся заболевания или синдромы. В качестве сопутствующего заболевания, которое может стимулировать синергетический эффект ИР, в данной модели использована вибрационная болезнь (ВБ), а фактором внешней среды, способствующего синергетическому эффекту, в данной модели принято электромагнитное поле (ЭП).

В качестве еще двух сегментов ФР выбраны традиционные факторы ИР (возраст, курение, артериальная гипертензия, сахарный диабет, избыточная масса тела, психосоциальный стресс, отягощенная наследственность) и дескрипторы, определенные на основе электрокардиографических исследований [7].

В базовой модели МИС, представленной на рис. 1, на первом иерархическом уровне строятся «слабые» классификаторы, решения которых агрегируются попарно в блоке синергетических классификаторов. Под воздействием внешних факторов (процесса обучения или экспертного оценивания), веса в нейронной сети NET приобретают оптимальные значения, то есть система самоорганизуется.

Процесс самоорганизации может привести к полному исключению из структуры системы АИА любого уровня или к синергетическому эффекту за счет усиления сочетанных факторов ИР. В выходном слое базовой модели используется нечеткая

нейронная сеть или однослойная нейронная сеть прямого распространения с линейной функцией активации.

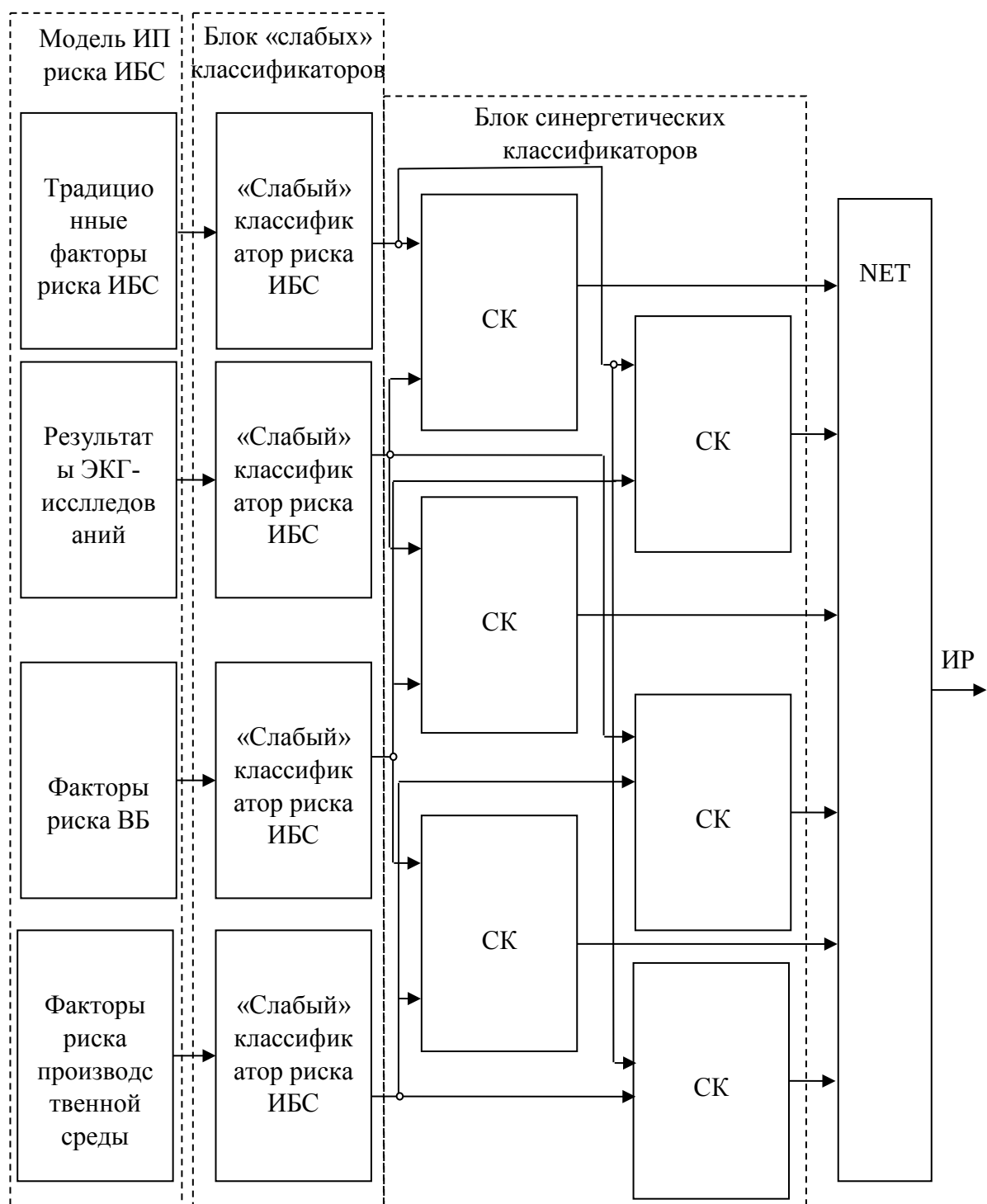


Рис.1. Структурная схема базовой модели мультиагентной интеллектуальной системы

Например, выход «слабого» классификатора по ФР «возраст» - x_1 формируется как коэффициент уверенности риска ИБС по x_1 по формуле:

$$KU_{ИБС}(x_1) = \frac{P_{ИБС}(фон) \cdot P(x_1|ИБС)}{P(x_1)}, \quad (1)$$

где $P_{ИБС}(фон)$ – априорная вероятность заболевания ИБС в экспериментальной группе, $P(x_1|ИБС)$ – условная вероятность возрастной группы с диагнозом ИБС, $P(x_1)$ – распределение экспериментальной группы по возрасту [7].

Полученные экспериментальные результаты показали, что разработанный метод прогнозирования медицинских рисков является универсальным инструментом. Решающие модули как «слабых», так и «сильных» классификаторов могут строиться на основе любой интеллектуальной платформы.

3. Обсуждение результатов

Для оценки качества прогноза риска ИР определялись диагностические показатели базовой модели классификатора с различными структурными решениями. Для оценки эффективности прогнозирования ИР базовой структуры классификатора проводились экспериментальные исследования показателей качества прогнозирования ИР моделями классификаторов, полученными из базовой модели с отключением определенных каналов принятия решений на различных иерархических уровнях. В качестве известного метода прогноза ИР, по сравнению с которым осуществлялось сопоставление показателей качества прогнозирования, выбрана шкала SCORE [1]. В качестве расчетных показателей качества диагностических решающих правил были выбраны показатели диагностической чувствительности (ДЧ), диагностической специфичности (ДС) и диагностической эффективности (ДЭ) [10].

Результаты работы моделей классификаторов по прогнозированию ИБС на контрольной выборке приведены на диаграммах рис. 2.

Для оценки эффективности включения в модель классификатора каналов побочных заболеваний и фактора производственной среды контрольная выборка формировалась из профессиональной группы машинистов электролокомотивов, для которых профессиональным заболеванием является ВБ, а ФР производственной среды – электромагнитное поле.

При построении не полной модели 1 отключались синергетические каналы, а в не полной модели 2 дополнительно не учитывались решения «слабых» классификаторов сопутствующих заболеваний и факторов производственной среды.

Анализ диаграмм рис. 2 показывает, что показатели качества полной модели классификатора выше на 8% показателей качества классификаторов с исключенными синергетическими каналами. При этом по основным показателям качества классификации: ДЧ, ДС и ДЭ предложенные модели принятия решений превосходят известные на 10...16%.

Полученные сравнительные характеристики экспертных оценок ССР и предлагаемых моделей МИС позволяют рекомендовать полученные технические и алгоритмические решения для практического использования в системах диагностики функционального состояния сердечно-сосудистой системы.

Выводы

1. Предложена базовая структура многоагентной интеллектуальной системы прогнозирования сердечно-сосудистых рисков, содержащая «сильные» и «слабые» классификаторы, при этом «слабые» классификаторы разделены на четыре группы, первая из которых осуществляет анализ данных, полученных на основе традиционных факторов риска ишемической болезни сердца, вторая – на основе анализа электрокардиологических исследований, третья группа «слабых» классификаторов предназначена для диагностики сопутствующих заболеваний и синдромов по

предикторам, используемых первыми двумя группами агентов, а четвертая – анализирует факторы риска окружающей среды, позволяющая управлять процессом принятия решений посредством сочетания экспертных оценок, статистических данных и текущей информации.

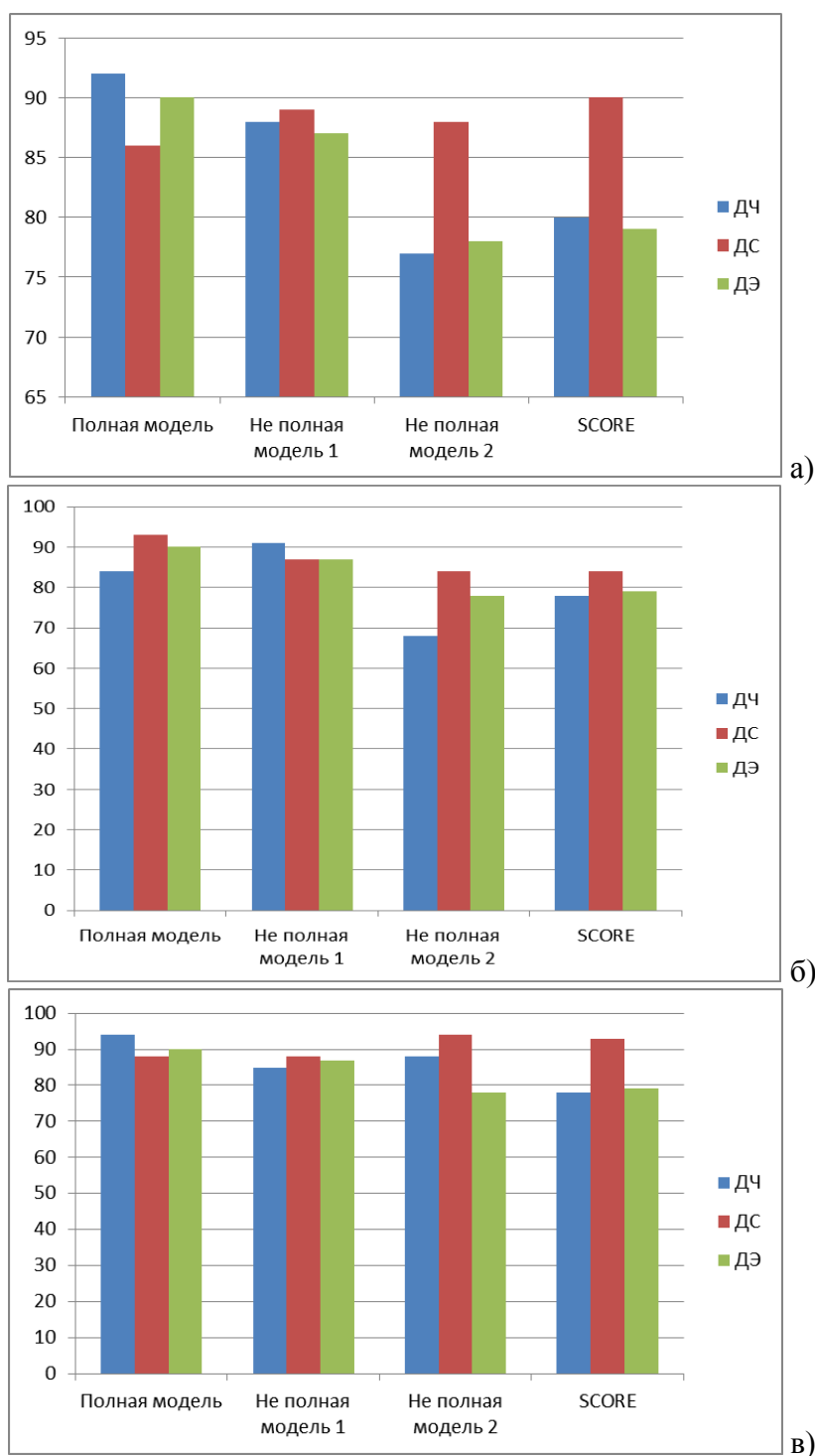


Рис.2. Диаграммы показателей качества классификаторов с различной структурой: а – для класса 1, б – для класса 2, в – для класса 3

2. Проведены экспериментальные исследования различных модификаций предложенной модели классификатора, заключающихся в последовательном отключении из агрегатора решений «слабых» классификаторов на различных иерархических уровнях. В ходе

экспериментального оценивания и в результате математического моделирования было показано, что при использовании всех информативных признаков уверенность в правильном прогнозе по ишемическому риску превышает величину 0,8 по всем группам наблюдения и по всем показателям качества классификации. Показатели качества прогнозирования выше, чем у известной системы прогнозирования SCORE, в среднем, на 14%.

Список литературы

1. Смирнова, М.Д. Прогностические факторы развития сердечно-сосудистых осложнений во время аномальной жары 2010 г. (кагорное наблюдательное исследование) / М.Д. Смирнова, Т.В. Фофанова, Ф.Т. Агеев [и др.] // Кардиологический вестник. — 2016. — №11(1). — С. 43-51.
2. Ефремов, М.А. Гибридные нечеткие модели для прогнозирования возникновения и осложнений артериальной гипертензии с учетом энергетических характеристик биоактивных точек / М.А. Ефремов, С.А. Филист, О.В. Шаталова [и др.] // Известия Юго-Западного государственного университета. Серия: Управление, вычислительная техника, информатика. Медицинское приборостроение. — 2018. — Т. 8, № 4 (29). — С. 104-119.
3. Филист, С.А. Использование гибридных нейросетевых моделей для многоагентных систем классификации в гетерогенном пространстве информативных признаков / С.А. Филист, А.Г. Курочкин, В.В. Жилин [и др.] // Прикаспийский журнал: управление и высокие технологии. Научно-технический журнал. — 2015. — № 3 (31). — С.85-95.
4. Филист, С.А. Универсальные сетевые модели для задач классификации биомедицинских данных / С.А. Филист, Р.А. Томакова, Зар До Яа // Известия Юго-Западного государственного университета. — 2012. — № 4 (43), Ч. 2. — С. 44-50.
5. Филист, С.А. Гибридная нейронная сеть с макрослоями для медицинских приложений / С.А. Филист, О.В. Шаталова, М.А. Ефремов // Нейрокомпьютеры. Разработка и применение. — 2014. — №6. — С. 35-39.
6. Филист, С.А. Модели нечетких нейронных сетей с трех-стабильным выходом в инструментарии для психологических и физиологических исследований / С.А. Филист, А.Р.С. Халед, О.В. Шаталова, В.В. Руденко // Системный анализ и управление в биомедицинских системах. — 2007. — Т.6, №2. — С. 475-479.
7. Комлев, И.А. Прогнозирование и оценка степени тяжести ишемии сердца на основе гибридных нечётких моделей / И.А. Комлев, О.В. Шаталова, С.В. Дегтярев, А.В. Серебровский // Известия Юго-Западного государственного университета. Серия Управление, вычислительная техника, информатика. Медицинское приборостроение. — 2019. — Т. 9, №1 (30). — С. 133-145.
8. Корневский, Н.А. Теория проектирования нечетких сетевых экспертных систем для управления медико-экологической безопасностью / Н.А. Корневский, С.А. Филист, А.Б. Красковский [и др.] // Научные ведомости Белгородского государственного университета. Серия: История. Политология. Экономика. Информатика. — 2009. — Т. 11, № 9-1 (64). — С.146-151.
9. Сидельников, Ю.В. Теория и организация экспертного прогнозирования / Ю.В. Сидельников. — М.: ИМЭМО АН СССР, 1990. — 196 с.
10. Омельченко, В.П. Практикум по медицинской информатике : учебное пособие для студ. мед. вузов России / В.П. Омельченко, А.А. Демидова. — Ростов на Дону: Феникс, 2001. — 304 с.

SYNTHESIS OF PARAMETERIZED FAMILIES OF CORRECTLY FUNCTIONING SIGMA-PI NEURONS

Shibzukhov Z.M.^{1,2}

¹ Institute of Mathematics and Informatics MPSU, Moscow, Russia

² Moscow Institute of Physics and Technology, Moscow, Russia, intellimath@mail.ru

The $\Sigma\Pi$ -neuron is biologically inspired formal model for logical information processing. The $\Sigma\Pi$ -neuron model adequately reflects the processes of information processing in the cerebral cortex [1,2] and in dendritic trees of neurons [3]. $\Sigma\Pi$ -neuron is able to represent arbitrary partial boolean function. The advantage of the $\Sigma\Pi$ -neuron model is the possibility of its constructive training (direct construction) in one pass of the training sample [4]. Using the constructive training procedure, it is also possible to build ensembles of $\Sigma\Pi$ -neurons. At the same time, each of them will function correctly on the training sample [5,6]. The $\Sigma\Pi$ -neuron model is very effective in correcting Boolean algorithms [7]. Effective electrical circuit design for $\Sigma\Pi$ -neurons was discussed in [8]. Using $\Sigma\Pi$ -neurons can reduce the number of layers in deep networks architecture [9].

The -neuron model define a boolean function in the following form:

$$y = \text{spn}(\mathbf{x}) = \text{sgn}(\text{sp}(\mathbf{x})),$$

where $y \in \{0,1\}$, $\mathbf{x} \in \{0,1\}^n$, $\text{sgn}(s) = \llbracket s \geq 0 \rrbracket^1$,

$$\text{sp}(\mathbf{x}) = \theta + \sum_{k=1}^M w_k \prod_{i \in \mathbf{i}_k} x_i, \quad (1)$$

$\theta \in \mathbb{R}$ is a free constant, $w_k \in \mathbb{R}$ is a weight of the k -th synaptic cluster formed by inputs with indices from the *multi-index* $\mathbf{i}_k \subset \{1, \dots, n\}$.

There is a constructive procedure for constructing (possibly large) ensembles of -neurons that function correctly on their training set. This is very useful when building majority-voting procedures that help to increase stability and generalization ability of the -neuron model. The possibility of parametric representation of such ensembles could allow us to solve the problems of searching for -neurons that meet certain criteria that can be formulated in terms of these parameters (for example, find -neurons with a potentially minimal number of multiplicative terms). It can also simplify the retraining process for new inputs.

In this paper, we propose a method for constructive training of parametric families of -neurons that function correctly on a training set. By setting the values of free parameters, you can get all -neurons from this family. In addition, it will be shown that the values of the weights of -neurons can be calculated using linear and threshold-linear neurons with integer weights, the input of which receives the values of free parameters. The example shows the possibility of easy additional training by setting the values of free parameters.

The problem

Let $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ be a finite set of Boolean n -dimensional vectors, $\mathbf{Y} = \{y_1, \dots, y_N\}$ be a set of Boolean values ($y_k \in \{0,1\}$) corresponding to the set $\mathbf{X}$. We consider Boolean functions $f(\mathbf{x})$ such that $y_k = f(\mathbf{x}_k)$ for all $k = 1, \dots, N$.

It is required to construct the representation of $f(\mathbf{x})$ in the form of a -neuron. Since for $N \ll 2^n$ there is a very large number of such representations, it is necessary to obtain some parametric description of this family, which would make it possible to extract -neurons from it that satisfy some optimization criteria.

To describe the algorithm for constructing a -neuron, we define the relation $\not\geq$ on the set of Boolean vectors.

¹ $\llbracket S \rrbracket = 1$ if S is true and $\llbracket S \rrbracket = 0$ if S is false.

Definition. For $\mathbf{x}', \mathbf{x}'' \in \{0,1\}^n$ and $\mathbf{i} \subseteq \{1, \dots, n\}$ we have relation $\mathbf{x}' \not\geq_{\mathbf{i}} \mathbf{x}''$ if there is $i \in \mathbf{i}$ such that $x'_i < x''_i$ (i.e., $x'_i = 0$ and $x''_i = 1$).

We define the relation of *triangular ordering* on the set of Boolean vectors.

Definition. The sequence $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ is triangularly ordered with respect to the sequence of multi-indices $\mathbf{I} = \{\mathbf{i}_1, \dots, \mathbf{i}_N\}$ if

- 1) for all $k = 1, \dots, N$ and all $i \in \mathbf{i}_k$ there is $x_{ki} = 1$;
- 2) for all pairs $j < k$: $\mathbf{x}_j \not\geq_{\mathbf{i}_k} \mathbf{x}_k$.

Fact. If $\mathbf{X}$ is ordered by increasing number of ones in the vector $\mathbf{x}$, then it is triangularly ordered with respect to $\mathbf{I}(\mathbf{X}) = \{\mathbf{i}(\mathbf{x}_1), \dots, \mathbf{i}(\mathbf{x}_N)\}$, where $\mathbf{i}(\mathbf{x}) = \{i: x_i = 1\}$.

The concept of triangular ordering has the following consequences when using the operator of the product of Boolean values.

Fact. If $\mathbf{X}$ is triangularly ordered with respect to $\mathbf{I}$, then

- 1) for all $k = 1, \dots, N$: $\prod_{i \in \mathbf{i}_k} x_{ki} = 1$;
- 2) for all pairs $j < k$: $\prod_{i \in \mathbf{i}_k} x_{ji} = 0$.

This fact serves as the basis for the direct construction of -neurons.

Construction of $\Sigma\Pi$ -neurons

Here is the constructive learning procedure for the -neuron, which is a slightly modified version of the standard learning procedure. To describe it, we introduce the notation of the poly-linear function at p -th step:

$$\text{sp}^p(\mathbf{x}) = \theta + \sum_{k=1}^p w_k \prod_{i \in \mathbf{i}_k} x_i.$$

The procedure for constructing a -neuron RCL-SPN (Recurrent Constructive Learning of SPN) from the sequence $\mathbf{X}$, triangularly ordered relative to $\mathbf{I}$, and the corresponding sequence of values $\mathbf{Y}$, has the form :

def RCL-SPN($\mathbf{I}, \mathbf{X}, \mathbf{Y}$):

$\text{sp}^0(\mathbf{x}) \leftarrow \theta$

for k **in** $1 \dots N$:

$s \leftarrow \text{sp}^{k-1}(\mathbf{x}_k)$

if $\text{sgn}(s) = y_k$:

$\text{sp}^k(\mathbf{x}) \leftarrow \text{sp}^{k-1}(\mathbf{x})$

else:

$w_k \leftarrow \tilde{y}_k - 1 - s$

$\text{sp}^k(\mathbf{x}) \leftarrow \text{sp}^{k-1}(\mathbf{x}) + w_k \prod_{i \in \mathbf{i}_k} x_i$

$\text{spn}(\mathbf{x}) \leftarrow \text{sgn}(\text{sp}^N(\mathbf{x}))$

return $\text{spn}(\mathbf{x})$

It is easy to establish that applying the RCL-SPN procedure to $\mathbf{X}, \mathbf{Y}$ we get correctly functioning -neuron $\text{spn}(\mathbf{x})$, i.e. for all $k = 1, \dots, N$: $y_k = \text{spn}(\mathbf{x}_k)$.

Construction of a parameterized family of -neurons

The set of all sequences of multi-indices with respect to which the sequence $\mathbf{X}$ is triangularly ordered, we denote $\mathcal{J}(\mathbf{X})$ and introduce the relation $<$.

Definition. $\mathbf{I}' < \mathbf{I}''$, if

- 1) for all $k = 1, \dots, N$: $\mathbf{i}'_k \subseteq \mathbf{i}''_k$;
- 2) there is k such that $\mathbf{i}'_k \subset \mathbf{i}''_k$.

Note that for any sequence $\mathbf{I} \in \mathcal{J}(\mathbf{X})$: $\mathbf{I} < \mathbf{I}(\mathbf{X})$ or $\mathbf{I} = \mathbf{I}(\mathbf{X})$.

Let $\mathcal{J}^*(\mathbf{X})$ denote the set of minimal elements in $\mathcal{J}(\mathbf{X})$ with respect to the relation $<$.

Fact. If $\{\mathbf{i}_1, \dots, \mathbf{i}_N\} \in \mathcal{J}^*(\mathbf{X})$, then for all $k = 1, \dots, N$ multi-index $\mathbf{i}_k \in \mathcal{J}_k(\mathbf{X})$, where $\mathcal{J}_k(\mathbf{X})$ is the set of minimal multi-indices $\mathbf{i} \subseteq \mathbf{i}(\mathbf{x}_k)$ such that

- 1) for all $i \in \mathbf{i}$: $x_{ki} = 1$;
- 2) for all $j < k$: $\mathbf{x}_j \not\geq_{\mathbf{i}} \mathbf{x}_k$.

For each $1 \leq k \leq N$ we define a multi-linear function

$$P_k(\mathbf{x}) = \sum_{\mathbf{i} \in \mathcal{I}_k(\mathbf{X})} \gamma_{\mathbf{i}} \prod_{i \in \mathbf{i}} x_i, \quad (2)$$

where $\gamma_{\mathbf{i}} \geq 0$ and

$$\sum_{\mathbf{i} \in \mathcal{I}_k(\mathbf{X})} \gamma_{\mathbf{i}} = 1. \quad (3)$$

Fact. $P_k(\mathbf{x})$ has the following properties:

1. for all $k = 1, \dots, N$: $P_k(\tilde{\mathbf{x}}_k) = 1$;
2. for all $j < k$: $P_k(\tilde{\mathbf{x}}_j) = 0$.

The set $\mathcal{I}_k(\mathbf{X})$ can be constructed using a procedure from .

Define a parameterized multi linear functions:

$$\text{sp}^p(\mathbf{x}) = \theta + \sum_{k=1}^p w_k P_k(\mathbf{x}), \quad 1 \leq p \leq N.$$

The following algorithm constructs a parameterized -neuron.

```

def RCL-SPN2(I, X, Y):
     $\text{sp}^0(\mathbf{x}) \leftarrow \theta$ 
    for  $k$  in  $1 \dots N$ :
         $s \leftarrow \text{sp}^{k-1}(\mathbf{x}_k)$ 
        if  $\text{sgn}(s) = y_k$ :
             $\text{sp}^k(\mathbf{x}) \leftarrow \text{sp}^{k-1}(\mathbf{x})$ 
        else:
             $w_k \leftarrow y_k - 1 - s$ 
             $\text{sp}^k(\mathbf{x}) \leftarrow \text{sp}^{k-1}(\mathbf{x}) + w_k P_k(\mathbf{x})$ 
     $\text{spn}(\mathbf{x}) \leftarrow \text{sgn}(\text{sp}^N(\mathbf{x}))$ 
    return  $\text{spn}(\mathbf{x})$ 

```

It is easy to establish that applying the RL-SPN2 procedure to $\mathbf{X}, \mathbf{Y}$ we obtain a parameterized -neuron $\text{spn}(\mathbf{x})$ such that for any set of parameter values for all $k = 1, \dots, N$: $y_k = \text{spn}(\tilde{\mathbf{x}}_k)$ under the condition (3) for all $\mathbf{i} \in \mathcal{I}_k(\mathbf{X})$.

Illustrative example

Consider the training set given by the table. In addition to the columns $x_1, \dots, x_4$ and y , which specify the values at the inputs and outputs, the column $\mathcal{I}_k$, which defines the minimum allowed multi-indices, and the column P_k , defining the corresponding multi-linear functions (2)–(3).

	x_1	x_2	x_3	x_4	y	$\mathcal{I}_k$	P_k
1	0	0	0	0	0		1
2	1	1	0	0	1	$\{1\}, \{2\}$	$ax_1 + (1-a)x_2$
3	0	0	1	1	1	$\{3\}, \{4\}$	$bx_3 + (1-b)x_4$
4	1	0	1	0	1	$\{1,3\}$	x_1x_3
5	0	1	0	1	1	$\{2,4\}$	x_2x_4
6	0	1	1	0	1	$\{2,3\}$	x_2x_3
7	1	0	0	1	1	$\{1,4\}$	x_1x_4

The sequence of multi-linear functions $\text{sp}_k(\mathbf{x})$ that are obtained during the construction process is as follows:

$$\begin{aligned}
& \text{sp}_1(\mathbf{x}) = -1 \\
& \text{sp}_2(\mathbf{x}) = \text{sp}_1(\mathbf{x}) + ax_1 + (1-a)x_2 \\
& \text{sp}_3(\mathbf{x}) = \text{sp}_2(\mathbf{x}) + bx_3 + (1-b)x_4 \\
& \text{sp}_4(\mathbf{x}) = \text{sp}_3(\mathbf{x}) + (1-a-b)_+x_1x_3.^1 \\
& \text{sp}_5(\mathbf{x}) = \text{sp}_4(\mathbf{x}) + (a+b-1)_+x_2x_4 \\
& \text{sp}_6(\mathbf{x}) = \text{sp}_5(\mathbf{x}) + (a+b-1)_+x_2x_4 \\
& \text{sp}_7(\mathbf{x}) = \text{sp}_6(\mathbf{x}) + (b-a)_+x_1x_4
\end{aligned}$$

Here $0 \leq a, b \leq 1$. As a result, we obtain a family of multi-linear functions:

$$\begin{aligned}
\text{sp}(\mathbf{x}; a, b) = & -1 + ax_1 + (1-a)x_2 + bx_3 + (1-b)x_4 + \\
& +(1-a-b)_+x_1x_3 + (a+b-1)_+x_2x_4 + \\
& +(a-b)_+x_2x_3 + (b-a)_+x_1x_4.
\end{aligned}$$

For clarity, we give several variants:

a	b	$\text{sp}(\mathbf{x})$
0	0	$-1 + x_2 + x_4 + x_1x_3$
0	1	$-1 + x_2 + x_3 + x_1x_4$
1	0	$-1 + x_1 + x_4 + x_2x_3$
1	1	$-1 + x_1 + x_3 + x_2x_4$
1	1	$-1 + x_1 + x_3 + x_2x_4$
$\frac{1}{2}$	$\frac{1}{2}$	$-1 + x_1 + x_3 + x_2x_4$

Since the minimum number of terms corresponds to combinations of Boolean values a and b , in the above table we can see such minimal -neurons.

It is easy to see that if the parameters a, b are considered as additional control parameters, then the -neuron takes the following form:

$$\begin{aligned}
\text{sp}(\mathbf{x}, \mathbf{u}) = & -1 + u_1x_1 + u_2x_2 + u_3x_3 + u_4x_4 + \\
& + u_5x_1x_3 + u_6x_2x_4 + u_7x_2x_3 + u_8x_1x_4,
\end{aligned}$$

where

$$u_1 = a, \quad u_2 = 1 - a, \quad u_3 = b, \quad u_4 = 1 - b, \quad (4)$$

$$u_5 = (1 - a - b)_+, \quad u_6 = (a + b - 1)_+, \quad u_7 = (a - b)_+, \quad u_8 = (b - a)_+. \quad (5)$$

As a result, we get a -neuron, in which the synaptic parameters are calculated using linear and ReLU control elements. In essence, it can be considered as single -neuron, which, in addition to the inputs $x_1, \dots, x_4$, also has inputs from an intermediate layer consisting of threshold-linear neurons (1)–(2).

In some cases it will be easy to adapt the parameterized -neuron without retraining on 2 new inputs. For example, let $(1, 0, 0, 0)$ be new input with expected value $y = 1$: $\text{sp}(1, 0, 0, 0; a, b) = -1 + ax_1$. So we may set $a = 1$ and get new parameterized -neuron with

¹ $(s)_+ = \max\{s, 0\}$.

$$\text{sp}(\mathbf{x}, b) = -1 + x_1 + bx_3 + (1 - b)x_4 + bx_2x_4 + (1 - b)x_2x_3,$$
where $0 \leq b \leq 1$. On next new input $(0,0,1,0)$ with $y = 1$: $\text{sp}(0,0,1,0; b) = -1 + b$. So we may set $b = 1$ and get

$$\text{sp}(\mathbf{x}) = -1 + x_1 + x_3 + x_2x_4.$$

Thus, parameterized $\Sigma\Pi$ -neurons can adapt to new inputs by setting values to the corresponding free control parameters.

Conclusion

Thus, we have built a constructive procedure that allows us to present a parametric representation for a family of $\Sigma\Pi$ -neurons that function correctly on the training set. This family is represented by a single $\Sigma\Pi$ -neuron with an intermediate layer of linear and threshold linear elements that calculate the weights of any $\Sigma\Pi$ -neuron from this family for given values of input control parameters. This is especially useful if the $\Sigma\Pi$ -neurons must participate in some sequential information processing process. The parametric representation of the $\Sigma\Pi$ -neuron family that functions correctly on the training set allows us to search for optimal combinations of control parameters within the framework of various optimization criteria. A simple illustrative example explains that it can also simplify the retraining process for new inputs.

References

1. Mel B.W. The sigma-pi column: a model of associative learning in cerebral cortex / B.W. Mel – Technical Report CNS Memo 6, Computation and Neural Systems Program, California Institute of Technology, 1990.
2. Mel B.W. An Augmented Two-Layer Model Captures Nonlinear Analog Spatial Integration Effects in Pyramidal Neuron Dendrites / M.P. Jadi, B.F. Behabadi, A. Poleg-Polsky, J. Schiller, B.W. Mel – In: Proceedings of the IEEE – 2014 – Vol. 102, No. 5 – PP. 782–798.
3. Mel B.W. Toward a simplified model of an active dendritic tree / B.W. Mel – In: Dendrites (Third edition), Chapter 16 – Oxford University Press – 2016 – PP. 405–486.
4. Timofeev A.V., Pshibikhov V.Kh. Learning Algorithms and Minimizing the Complexity of Polynomial Recognizing Systems / A.V. Timofeev, V.Kh. Pshibikhov – Izv. AN SSSR. Technical cybernetics – 1974 – No. 7 – PP. 214–217 (in russian).
5. Shibzukhov Z.M. Constructive learning methods of $\Sigma\Pi$ -neural networks / Z.M. Shibzukhov – 2006 – MAIK Nauka – 150 p. (in russian).
6. Shibzukhov Z.M. Correct Aggregation Operations with Algorithms / Z.M. Shibzukhov – Pattern Recognition and Image Analysis – 2014 – Vol. 24, No. 3 – PP. 377–382.
7. Lyutikova L.A. Sigma-Pi Neural Networks: Error Correction Methods / L.A. Lyutikova – Procedia Computer Science – 2018 – Vol. 145 – PP.312–318.
8. Potapov A.A. Sigma-pi neural unit: switching circuit simulation / A.A. Potapov, W. Zhang, T. Feng, S.Sh. Rekhviashvili – AIIPCC '19: Proceedings of the International Conference on Artificial Intelligence, Information Processing and Cloud Computing – 2019 – Article No. 21 – PP. 1–5. <https://doi.org/10.1145/3371425.3371477>
9. Delalleau O. Shallow vs. Deep Sum-Product Networks / O. Delalleau, Y. Bengio – Advances in Neural Information Processing Systems 24 – 2011 – PP. 666–674.

СРАВНЕНИЕ НЕКОТОРЫХ МЕТОДОВ ОБРАБОТКИ ДАННЫХ В ЗАДАЧЕ АНАЛИЗА ДЛИТЕЛЬНЫХ ИЗМЕРЕНИЙ ТЕМПЕРАТУРЫ

О.С.Володько¹, Н.А.Буряк², Л.А.Компаниец¹

¹Институт вычислительного моделирования СО РАН, osv@icm.krasn.ru

²Сибирский федеральный университет,
Институт математики и фундаментальной информатики

Методы обработки экспериментальных данных начали разрабатываться уже более двух веков назад для решения практических задач в медицине, агробиологии, экономике, социологии.

Наиболее простым способом обработки данных является вычисление среднего арифметического значения. Но, несмотря на его простоту, этот метод не срабатывает при большом диапазоне значений, особенно, если в данных содержатся как положительные, так и отрицательные величины.

Не менее распространенным методом, хотя и более сложным для реализации, является метод наименьших квадратов [1]. Достоинство этого метода в том, что все вычисления сводятся к нахождению неизвестных коэффициентов аппроксимирующей функции. Недостаток в том, что этот метод чувствителен к резким выбросам в данных. Кроме того, довольно сложно подобрать такую аналитическую функцию, которая бы требуемым образом аппроксимировала исходные данные.

Обобщением метода наименьших квадратов является метод главных компонент, задача которого в том, чтобы получить максимум информации, используя минимальный набор данных [2]. Недостатком этого метода является то, что данный метод не учитывает нелинейные зависимости между исследуемыми величинами, но эти зависимости можно попробовать выявить с помощью нелинейного метода главных компонент, например, метода главных кривых [3].

В данной работе метод главных кривых был применен к анализу данных гидрофизических измерений водоема (озеро Шира, Хакасия) и было проведено сравнение результатов его работы с другими методами.

В результате натурных измерений данные температуры получают с некоторой погрешностью, что происходит в силу турбулентного характера течения в озере и приближенного измерения температуры приборами.

Необходимость уточнения характера изменений температуры вызвана тем, что по этим данным определяются такие важные характеристики течения как области прохождения внутренних волн [4].

В 2015 году на озере Шира были проведены длительные измерения температуры в разных точках озера, указанных на рис. 1, с помощью термисторных цепей. С их помощью проводятся измерения температуры в указанных местоположениях посредством записи данных с определенным интервалом времени.

Для нашей задачи были проанализированы данные в шести точках: T1, T2, T3, T4, T5, T6, с глубинами 14.4, 18.7, 16.68, 23.1, 18.7, 12.68 м соответственно. Измерения проводились с промежутком 0.5 минуты на 23-31 горизонтах по глубине с интервалом 0.5-1.5 м по вертикали между датчиками. Был взят промежуток времени в 1 час, с 12:00 до 13:00 15.06.2015.



Рис. 1. Расположение термисторных цепей в 2015 году.

На рис. 2 представлены профили температуры в точках T1, T2, T3, T4, T5 и T6, построенные по исходным данным и после применения метода среднего арифметического.

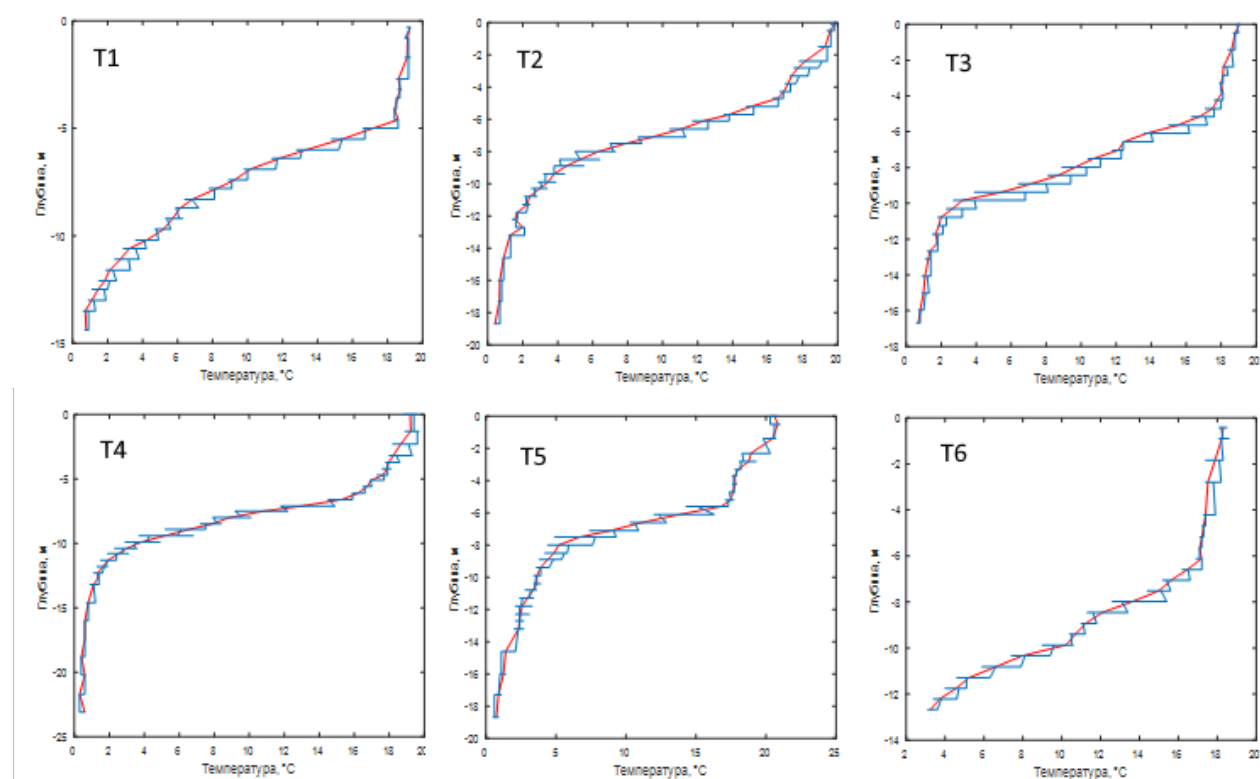


Рис. 2. Средние значения температуры в точках T1, T2, T3, T4, T5, T6.
Красная линия – средние значения температуры, синяя линия – исходные данные температуры.

Метод наименьших квадратов [5] был применен к анализу длительных измерений температуры на озере Шира с использованием многочленов первой, второй и третьей степени (рис. 3).

В случае, когда берется аналитическое выражение в виде многочлена первой степени, метод наименьших квадратов совпадает с методом главных компонент [2].

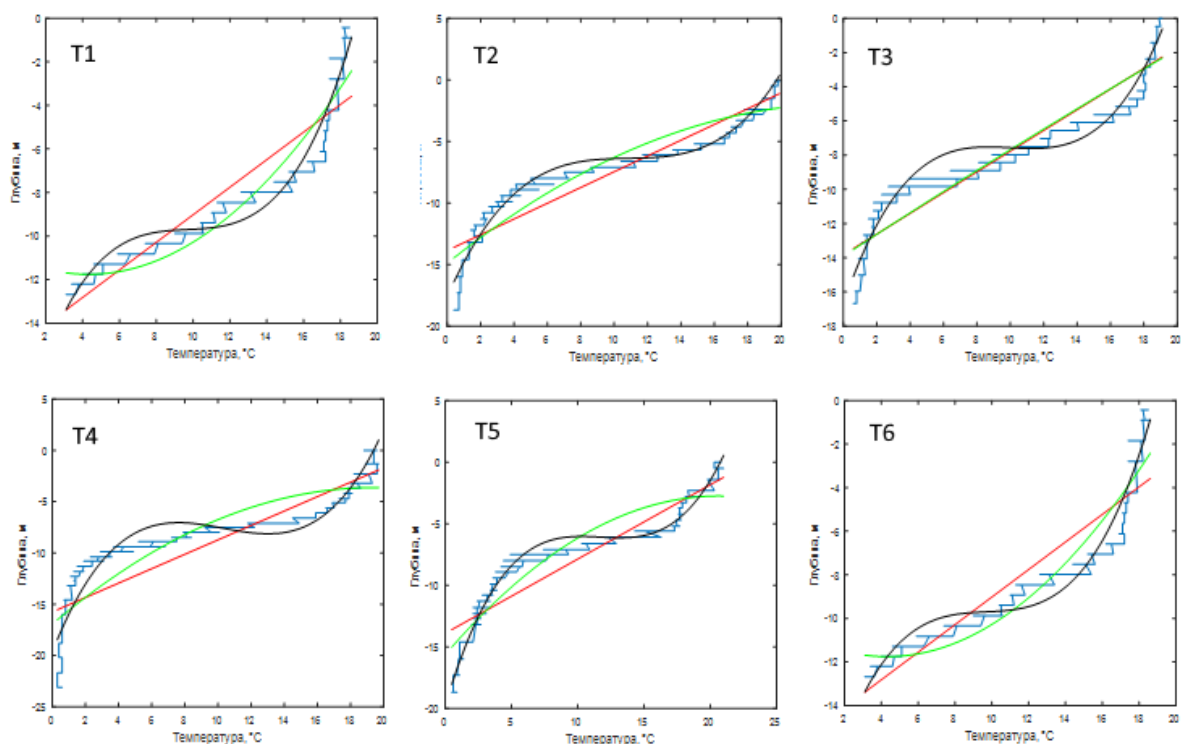


Рис. 3. Профили температуры, полученные методом наименьших квадратов в точках T1, T2, T3, T4, T5, T6. Красная линия – полином первого порядка, зеленая линия – полином второго порядка, черная линия – полином третьего порядка, синяя линия – исходные данные температуры.

Кроме полиномов, в качестве аппроксимирующей функции может быть использована любая аналитически заданная функция. Одним из вариантов такой функции, используемой при трехмерном численном моделировании озера Шира в летний период, является функция вида:

$$y = a + b \operatorname{th}(-kz - z_0), \quad (1)$$

где z – глубина, a , b , k , z_0 – параметры, определяемые методом наименьших квадратов.

Результаты применения метода наименьших квадратов для этой функции представлены на рис. 4. Значения параметров для каждой из точек T1, T2, T3, T4, T5, T6 представлены в Таблице.

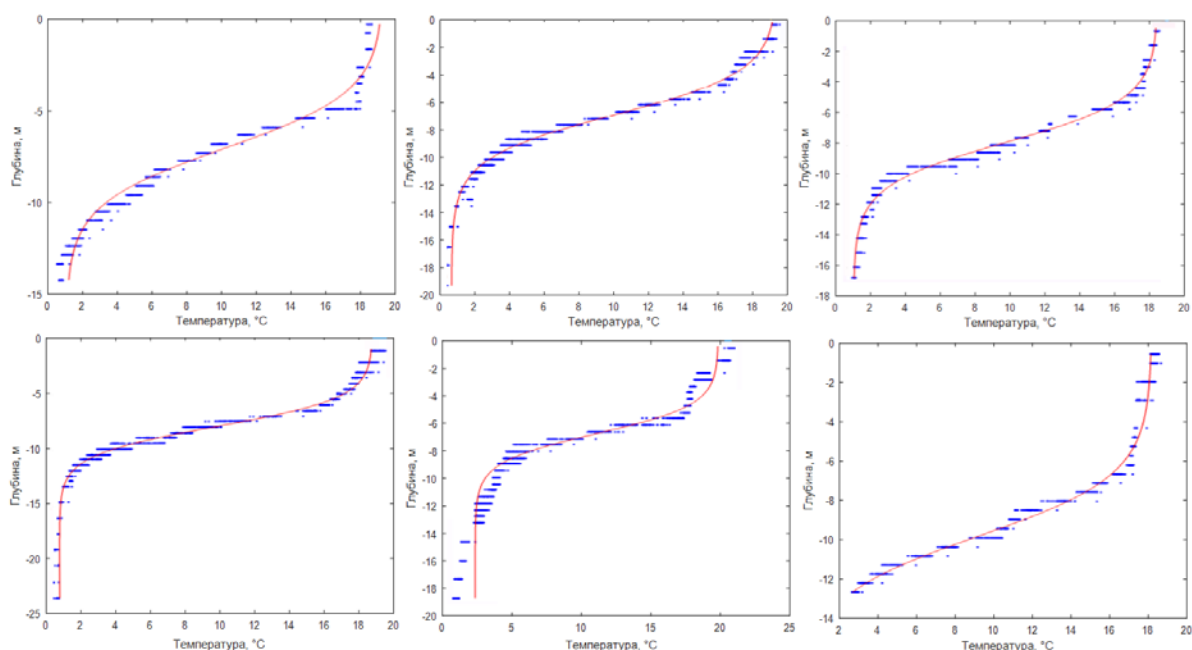


Рис. 4. Профили температуры в точках T1, T2, T3, T4, T5, T6, полученные методом наименьших квадратов с использованием функции (1). Красная линия – график функции (1), синяя линия – исходные данные температуры.

Таблица

Значения параметров для метода наименьших квадратов
в точках T1, T2, T3, T4, T5, T6

	T1	T2	T3	T4	T5	T6
a	10.56	10.29	9.756	9.81	10.97	9.423
b	-9.475	-9.343	-9.107	-9.147	-8.88	8.711
k	-0.3226	-0.3286	-0.35	-0.4015	-0.4809	0.3147
z_0	2.302	2.238	2.728	3.196	3.226	-3.108
Доверительный интервал 95%	(10.52; 10.6)	(10.26; 10.31)	(9.738; 9.791)	(9.788; 9.832)	(10.93; 11.01)	(9.348; 9.498)

Главные кривые являются нелинейным обобщением главных компонент, так как в качестве начального приближения используется прямая линия, которая обычно является первой главной компонентой. Метод применяется для того, чтобы уменьшить размерность данных, для визуализации и сжатия данных. В литературе много определений главных кривых. Приведем одно из первых, данное в статье Hastie [3]. Оно

заключается в том, чтобы определить главные кривые определенной длины, которые достигают минимального ожидаемого квадрата расстояния от точек до их проекции на кривую. Для нашей задачи мы применили один из способов реализации метода главных кривых, а именно метод k -сегментов, описанный в статье Verbeek [6]. Его программная реализация представлена в [7].

В данном методе мы формируем полигональные линии путем нахождения и соединения таких отрезков, у которых квадрат расстояния от всех точек данных до этих отрезков является минимальным.

Метод главных кривых представлен на рис. 5

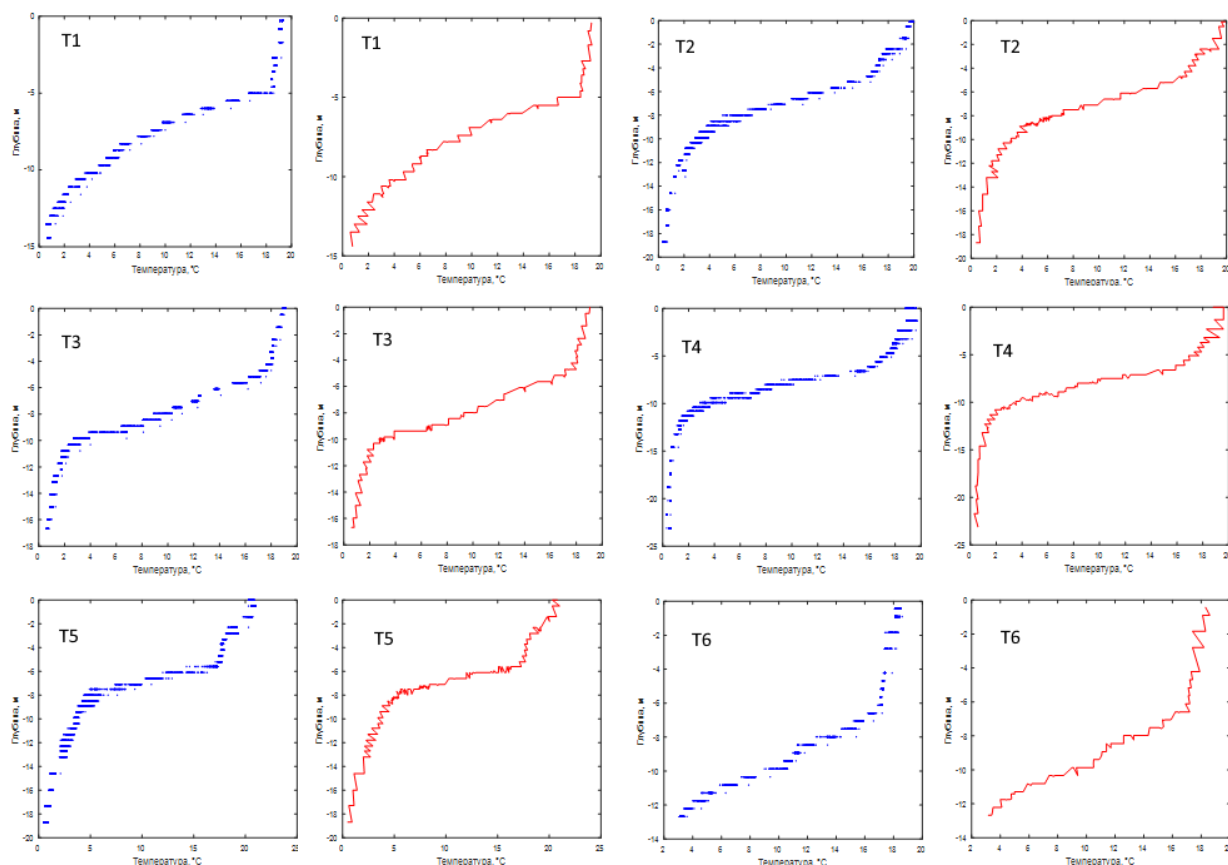


Рис. 5. Профили температуры в точках T1, T2, T3, T4, T5, T6, полученные по методу главных кривых – красные линии, исходные данные температуры – синие линии.

Если рассматривать данные измерений температуры во всех шести точек озера как одно измерение, то получим результат, представленный на рис. 6.

В данном случае применение метода главных кривых не привело к удовлетворительному результату, что, по-видимому, вызвано сильным разбросом в исходных данных измерений температуры. Один из возможных методов обработки сильно разбросанных временных данных представлен в статье [8].

Были проведены расчеты при разном объеме исходных данных. Метод k -сегментов требует $O(kn^2)$ итераций, где k – количество сегментов, n – количество

измерений, поэтому при необходимости анализа большого объема данных требуется большое количество памяти. Например, для нашей двумерной задачи, при количестве измерений температуры более 80000 (24 часа, измерение раз в полминуты), проведение расчета было возможно при наличии свободной оперативной памяти объемом 150 Гб. Если для поставленной задачи требуется проанализировать большой объем исходных данных, то данное ограничение можно обойти, используя метод, описанный в статье [9], который требует $O(k^2n)$ итераций.

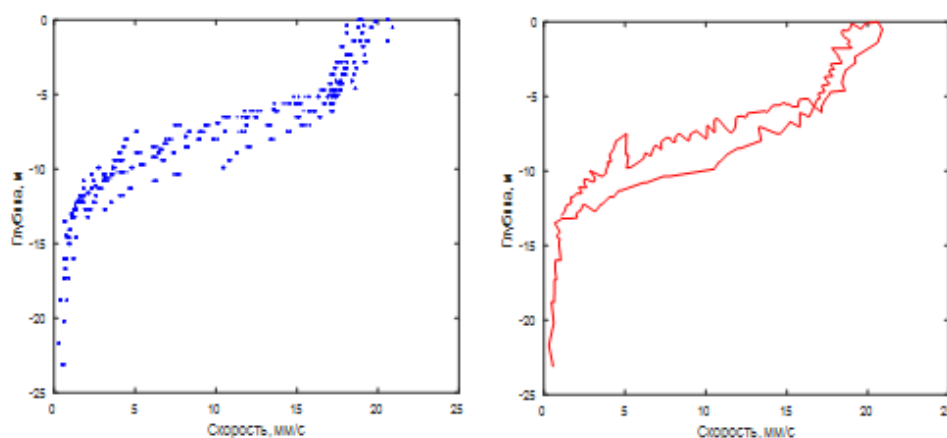


Рис. 6. Синие линии слева – исходные данные температуры, красная линия справа – профиль температуры, полученной по методу главных кривых.

Сравнение нескольких методов показывает, что метод главных кривых дополняет известные методы исследования. В отличие от метода наименьших квадратов и метода главных компонент получаемые с его помощью значения всегда находятся внутри интервала измеренных величин на тех горизонтах, на которых происходят измерения. Такой же особенностью обладает метод среднего арифметического, но он сильно сглаживает картину изменения температуры по глубине, лишая тем самым возможности определения глубин, на которых имеется немонотонное изменения температуры по глубине, т.е. участков перемешивания жидкости по глубине.

Список литературы

1. Линник Ю.В. Метод наименьших квадратов и основы математико-статистической теории обработки наблюдений / Ю.В.Линник // Гос. изд-во физико-математической литературы, 1958.
2. Jolliffe I.T. Principal Component Analysis / I.T.Jolliffe — Series: Springer Series in Statistics, 2nd ed., Springer, NY, XXIX, 2002 — 487 p.
3. Hastie T. Principal curves / T.Hastie, W.Stuetzle // Journal of the American Statistical Association. — 1989. — V. 84. — №. 406. — P. 502-516.
4. Hutter K., ed. Nonlinear internal waves in lakes / K.Hutter — Springer Science & Business Media, 2011 — V. 2.
5. Березин И.С. Методы вычислений / И.С.Березин, Н.П.Жидков // Гос. изд-во физико-математической литературы, 1962. — Т. 2.
6. Verbeek J.J. A k-segments algorithm for finding principal curves / J.J.Verbeek, N.Vlassis, B.Kröse // Pattern Recognition Letters. — 2002. — V. 23. — №. 8. — P. 1009-1017.

7. Homepage J.J.Verbeek. [Электронный Ресурс] URL: <http://thoth.inrialpes.fr/~verbeek/software.php> (дата обращения 15.09.2020).
8. Verbeek J.J. A soft k-segments algorithm for principal curves //International Conference on Artificial Neural Networks / J.J.Verbeek., N.Vlassis, B.Kröse // Springer, Berlin, Heidelberg — 2001. — P. 450-456.
9. Magwene P.M. Reconstructing the temporal ordering of biological samples using microarray data / P. M.Magwene, P.Lizardi, J.Kim // Bioinformatics. — 2003. — V. 19. — №. 7. — P. 842-850.

СПИСОК ДОКЛАДОВ

А.В.Бахшиев, И.С. Фомин, Ф.В.Гунделах, А.А. Демчева, А.М. Корсаков АРХИТЕКТУРА ПРОГРАММНОЙ ПЛАТФОРМЫ ДЛЯ РАЗРАБОТКИ СРЕДСТВ МОДЕЛИРОВАНИЯ РАСТУЩИХ СПАЙКОВЫХ НЕЙРОННЫХ СЕТЕЙ	3
Н.М. Богатов, Л.Р. Григорьян, А.С. Синицын, С.К. Шаповалов ОПРЕДЕЛЕНИЕ ТЕМПЕРАТУРЫ МОЗГА ПО МАГНИТНО-РЕЗОНАНСНЫМ ТОМОГРАФИЧЕСКИМ ИЗОБРАЖЕНИЯМ	11
В.С.Воронов, А.И.Рубан ИДЕНТИФИКАЦИЯ МОДЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ АНАЛОГОВ ФУНКЦИЙ ЧУВСТВИТЕЛЬНОСТИ	18
С.Е. Головенкин, С.Ю. Никулина, Ю.В. Орлова, Е.Ю. Пелипецкая, М. Г. Доррер ПОДБОР АРХИТЕКТУРЫ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ ДЛЯ ПРЕДСКАЗАНИЯ ПОСЛЕДСТВИЙ ИНФАРКТА МИОКАРДА	24
С.Е. Головенкин, С.Ю. Никулина, Ю.В. Орлова, Е.Ю. Пелипецкая, М. Г. Доррер ОЦЕНКА ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ПРЕДСКАЗАНИЯ ОСЛОЖНЕНИЙ ИНФАРКТА МИОКАРДА	31
М.В. Гунер, И.А. Бусс РАЗРАБОТКА СИСТЕМЫ РАСПОЗНАВАНИЯ КОРОВ И ОПРЕДЕЛЕНИЯ УРОВНЯ ИХ АКТИВНОСТИ НА ОСНОВЕ ПРИМЕНЕНИЯ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ	38
А.Ю. Дорогов КЛАСТЕРНЫЙ АНАЛИЗ ВЫСОКОРАЗМЕРНЫХ ДАННЫХ В ЗАДАЧЕ КЛАССИФИКАЦИИ ОБРАЗОВ	42
М. Г. Доррер, А.В. Фомин, Д.А. Логинов КЛАСТЕРИЗАЦИЯ ПОКУПАТЕЛЕЙ СИСТЕМЫ МАКСБОНУС С ПРИМЕНЕНИЕМ САМООРГАНИЗУЮЩИХСЯ КАРТ КОХОНЕНА	50
М. Г. Доррер, А. Е. Толмачева СРАВНЕНИЕ ЭФФЕКТИВНОСТИ РАБОТЫ АРХИТЕКТУР YOLOV3 И MASC R-CNN В КОМПЬЮТЕРНОМ ЗРЕНИИ УМНОГО ХОЛОДИЛЬНИКА	57

Л.Н. Елисов, Н.И. Овченков, В.И. Горбаченко, И.А. Абрамов НЕЙРОСЕТЕВАЯ КЛАССИФИКАЦИЯ АВИАЦИОННОГО ПЕРСОНАЛА КАК ЭЛЕМЕНТ ИНФОРМАЦИОННО- УПРАВЛЯЮЩЕГО ПРОСТРАНСТВА БЕЗОПАСНОСТИ ОБЪЕКТА ТРАНСПОРТНОЙ ИНФРАСТРУКТУРЫ	66
С.В.Листопад МЕТОД ОЦЕНКИ СПЛОЧЕННОСТИ ГИБРИДНОЙ ИНТЕЛЛЕКТУАЛЬНОЙ МНОГОАГЕНТНОЙ СИСТЕМЫ. ЧАСТЬ 2: СХОДСТВО ОНТОЛОГИЙ АГЕНТОВ	73
Н.М. Луговая, А.С. Михалев, Т.Г. Пенькова ПРОГНОЗИРОВАНИЕ ПОКАЗАТЕЛЕЙ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА ПРОИЗВОДСТВА АЛЮМИНИЯ НА ОСНОВЕ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ	79
М.Е. Марков, В.В. Варлашин, А.В. Бахшиев, В.Н. Уланов СРАВНИТЕЛЬНЫЙ АНАЛИЗ НЕЙРОСЕТЕВЫХ КЛАССИФИКАТОРОВ ДЛЯ РАСПОЗНАВАНИЯ ФАЗ ПОХОДКИ ЧЕЛОВЕКА	85
Б.В.Олейников, Н.И.Кожевников ЧИСЛЕННОЕ ИССЛЕДОВАНИЕ ОСНОВАННОГО НА МСТS МЕТОДА MOTIFS ДЛЯ ОТБОРА ПРИЗНАКОВ	91
Б.В.Олейников, А.В.Стрельников ОПРЕДЕЛЕНИЕ ОПТИМАЛЬНОГО НАБОРА ПРИЗНАКОВ ДЛЯ ЗАДАЧИ LSB-СТЕГОАНАЛИЗА АУДИОФАЙЛОВ ФОРМАТА WAVE	97
И.С. Пичковский, В.Е. Зобов КЛАСТЕРИЗАЦИЯ ПОСРЕДСТВОМ КВАНТОВОГО ОТЖИГА НА КВАНТОВЫХ ТРЁХУРОВНЕВЫХ ЭЛЕМЕНТАХ – КУТРИТАХ	103
А.А. Трифонов, А.А. Кузьмин, С.А. Филист, Е.В. Петрунина НЕЙРОСЕТЕВАЯ МОДЕЛЬ ДЛЯ УПРАВЛЕНИЯ ВЕРТИКАЛИЗАЦИЕЙ ЭКЗОСКЕЛЕТА В КОМБИНИРОВАННОМ РЕЖИМЕ	110
И.С. Фомин, А.В.Бахшиев ОБЛАЧНАЯ СИСТЕМА ПОЛУАВТОМАТИЧЕСКОЙ РАЗМЕТКИ НАБОРОВ ДАННЫХ ДЛЯ ОБУЧЕНИЯ ГЛУБОКИХ НЕЙРОННЫХ СЕТЕЙ	116
Л.В.Хливненко, Ф.А.Пятакович НЕЙРОСЕТЕВОЙ МЕТОД АДАПТАЦИИ КОНТЕНТА В ЗАВИСИМОСТИ ОТ СОСТОЯНИЯ ПОЛЬЗОВАТЕЛЕЙ	122

О.В. Шаталова, Д.А. Медников, З.У. Протасова, Н.С. Стадниченко ПРОГНОЗ РИСКА СЕРДЕЧНО-СОСУДИСТЫХ ОСЛОЖНЕНИЙ С СЕГМЕНТИРОВАННЫМ ПРОСТРАНСТВОМ ФАКТОРОВ РИСКА И СИНЕРГЕТИЧЕСКИМИ КАНАЛАМИ	126
Shibzukhov Z.M. SYNTHESIS OF PARAMETERIZED FAMILIES OF CORRECTLY FUNCTIONING SIGMA-PI NEURONS	132
О.С.Володько, Н.А.Буряк, Л.А.Компаниец СРАВНЕНИЕ НЕКОТОРЫХ МЕТОДОВ ОБРАБОТКИ ДАННЫХ В ЗАДАЧЕ АНАЛИЗА ДЛИТЕЛЬНЫХ ИЗМЕРЕНИЙ ТЕМПЕРАТУРЫ	138

СПИСОК АВТОРОВ

Ф.И.О.	Место работы	Стр.
Абрамов Игорь Анатольевич	440026, Пенза, ул. Красная, 40, Пензенский государственный университет E-mail: igora379@gmail.com	66
Бахшиев Александр Валерьевич	Центральный научно-исследовательский и опытно-конструкторский институт робототехники и технической кибернетики (ЦНИИ РТК) E-mail: alexab@rtc.ru	3, 85, 116
Богатов Николай Маркович	350040, г. Краснодар, ул. Ставропольская, 149, ФГБОУ ВПО «Кубанский государственный университет», E-mail: bogatov@phys.kubsu.ru	11
Буряк Никита Александрович	660041, Красноярск, Свободный пр., 79 корпус 3, Сибирский федеральный университет, Институт математики и фундаментальной информатики E-mail: buryakna98@gmail.com	138
Бусс И.А.	656038, Российская Федерация, Алтайский край, г. Барнаул, пр. Ленина, 46, Алтайский государственный технический университет им. И. И. Ползунова, E-mail: horyzont1@mail.ru	38
Варлашин Виктор Витальевич	195251, Санкт-Петербург, Политехническая ул., 29, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ) E-mail: varlashin_vv@spbstu.ru	85
Володько Ольга Станиславовна	660036, Красноярск, Академгородок, Институт вычислительного моделирования СО РАН E-mail: ovs@icm.krasn.ru	138
Воронов Владимир Сергеевич	660074, Красноярск, ул. Академика Киренского, д.26, Институт космических и информационных технологий (ИКИТ СФУ) E-mail: voronov.volodya2013@yandex.ru	18
Головенкин Сергей Евгеньевич	660022, Красноярск, ул. Партизана Железняка, 1, Красноярский государственный медицинский университет. имени профессора В.Ф.Войно-Ясенецкого E-mail: gse2008@mail.ru	24, 31
Горбаченко Владимир Иванович	440026, Пенза, ул. Красная, 40, Пензенский государственный университет E-mail: gorvi@mail.ru	66
Григорьян Леонтий Рустемович	350040, Краснодар, ул. Ставропольская 149, Кубанский государственный университет E-mail: leonmezon@mail.ru	11
Гунделах Филипп Викторович	195251, Санкт-Петербург, Политехническая ул., 29, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ) E-mail: f.gundelakh@yandex.ru	3

Гунер Михаил Викторович	656038, Российская Федерация, Алтайский край, г. Барнаул, пр. Ленина, 46, Алтайский государственный технический университет им. И. И. Ползунова, E-mail: horyzont1@mail.ru	38
Демчева Александра Андреевна	195251, Санкт-Петербург, Политехническая ул., 29, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ) E-mail: alexab@rtc.ru	3
Доррер Михаил Георгиевич	660000, Красноярск, просп. имени газеты Красноярский Рабочий, 31, Сибирский государственный университет науки и технологий им. М.Ф. Решетнева E-mail: dorrer_mg@sibsau.ru	24, 31, 50, 57
Дорогов Александр Юрьевич	197376, Санкт-Петербург, ул. проф. Попова д. 5, Санкт-Петербургский государственный электротехнический университет "ЛЭТИ" E-mail: vaksa2006@yandex.ru	42
Елисов Лев Николаевич	125438, Москва, ул. Михалковская, д.67, корп.1, ФГУП Государственный научно-исследовательский институт гражданской авиации, Министерство транспорта Российской Федерации, Москва, Российская Федерация E-mail: lev.el@list.ru	66
Зобов Владимир Евгеньевич	660036, Красноярск, ул. Академгородок, 50, стр. 43, Институт физики им. Л.В. Киренского Сибирского отделения Российской академии наук E-mail: rsa@iph.krasn.ru	103
Кожевников Никита Игоревич	660041, Красноярск, Свободный пр., 82А, Сибирский федеральный университет, Институт математики и фундаментальной информатики E-mail: nkozhevnikov@outlook.com	91
Компаниец Лидия Алексеевна	660036, Красноярск, Академгородок, Институт вычислительного моделирования СО РАН E-mail: kla@icm.krasn.ru	138
Корсаков Антон Михайлович	Центральный научно-исследовательский и опытно-конструкторский институт робототехники и технической кибернетики (ЦНИИ РТК) E-mail: alexab@rtc.ru	3
Кузьмин Александр Алексеевич	305040, г. Курск, ул. 50 лет Октября, 94, Юго-Западный государственный университет E-mail: ku3bmin@gmail.com	110
Листопад Сергей Викторович	Калининград, ул. Гостиная, д.5, Калининградский филиал ФИЦ "Информатика и управление" РАН E-mail: ser-list-post@yandex.ru	73
Логинов Дмитрий Александрович	ООО «Фабрика решений» E-mail: a.v.fomin@live.com	50
Луговая Нина Михайловна	Сибирский федеральный университет E-mail: luninok@gmail.com	79
Марков Максим Евгеньевич	195251, Санкт-Петербург, Политехническая ул., 29, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ)	85

	E-mail: maxxmarkov@gmail.com	
Медников Дмитрий Андреевич	305040, г. Курск, 0 лет Октября ул., 94, Юго-Западный государственный университет E-mail: kstu-bmi@yandex.ru	126
Михалев Антон Сергеевич	Сибирский федеральный университет E-mail: asmikhalev@yandex.ru	79
Никулина Светлана Юрьевна	660022, Красноярск, ул. Партизана Железняка, 1, Красноярский государственный медицинский университет. Имени профессора В.Ф.Войно-Ясенецкого E-mail: gse2008@mail.ru	24, 31
Овченков Николай Иванович	150003, Ярославль, Советская ул., 14, Ярославский государственный университет им. П.Г. Демидова, E-mail: ovchenkov@electronika.ru	66
Олейников Борис Васильевич	660041, Красноярск, Свободный пр., 82А, Сибирский федеральный университет, Институт математики и фундаментальной информатики E-mail: oleynik48@mail.ru	91, 97
Орлова Юлия Викторовна	660022, Красноярск, ул. Партизана Железняка, 1, Красноярский государственный медицинский университет. Имени профессора В.Ф.Войно-Ясенецкого E-mail: gse2008@mail.ru	24, 31
Пелипецкая Елена Юрьевна	660022, Красноярск, ул. Партизана Железняка, 1, Красноярский государственный медицинский университет. Имени профессора В.Ф.Войно-Ясенецкого E-mail: gse2008@mail.ru	24, 31
Пенькова Татьяна Геннадьевна	660036, Красноярск, Академгородок, Институт вычислительного моделирования СО РАН E-mail: penkova_t@icm.krasn.ru	79
Протасова Зейнаб Усама	305040, г. Курск, 0 лет Октября ул., 94, Юго-Западный государственный университет E-mail: zeinab.zeidan@yandex.ru	126
Петрунина Елена Валерьевна	107150, Москва, Московский государственный гуманитарно-экономический университет E-mail: petrunina@mggeu.ru	110
Пичковский Иван Сергеевич	660036, Красноярск, ул. Академгородок, 50, стр. 43, Институт физики им. Л.В. Киренского Сибирского отделения Российской академии наук E-mail: rsa@iph.krasn.ru	103
Пятакович Феликс Андреевич	308015, Белгород, ул. Победы, 85, ФГАОУ ВО Белгородский государственный национальный исследовательский университет E-mail: piatakovich@gmail.com	122
Рубан Анатолий Иванович	660074, Красноярск, ул. Академика Киренского, д.26, Институт космических и информационных технологий (ИКИТ СФУ) E-mail: ai-rouban@mail.ru	18
Синицын Алексей Сергеевич	350040, Краснодар, ул. Ставропольская 149, Кубанский государственный университет	11

	E-mail: nj_1@inbox.ru	
Стадниченко Никита Сергеевич	305040, г. Курск, 0 лет Октября ул., 94, Юго-Западный государственный университет E-mail: stadnikita@gmail.com	126
Стрельников Андрей Владимирович	660041, Красноярск, Свободный пр., 82А, Сибирский федеральный университет, Институт математики и фундаментальной информатики E-mail: avstrelnikov95@gmail.com	97
Толмачева А.Е.	ООО «Фабрика решений» E-mail: vkarmeo@mail.ru	57
Трифонов Андрей Андреевич	305040, г. Курск, ул. 50 лет Октября, 94, Юго-Западный государственный университет E-mail: voldraf@mail.ru	110
Уланов Владимир Николаевич	195251, Санкт-Петербург, Политехническая ул., 29, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ) E-mail: ulanov_vn@spbstu.ru	85
Филист Сергей Алексеевич	305040, г. Курск, ул. 50 лет Октября, 94, Юго-Западный государственный университет E-mail: Sfilist@gmail.com	110
Фомин Анатолий Владимирович	ООО «Фабрика решений» E-mail: a.v.fomin@live.com	50, 116
Фомин Иван Сергеевич	Центральный научно-исследовательский и опытно-конструкторский институт робототехники и технической кибернетики (ЦНИИ РТК) E-mail: i.fomin@rtc.ru	3
Хливненко Любовь Владимировна	39400, Воронеж, ул. 20 лет Октября, 84, ФГБОУ ВО Воронежский государственный технический университет E-mail: hlivnenko_lv@mail.ru	122
Шаповалов Сергей Константинович	350040, Краснодар, ул. Ставропольская 149, Кубанский государственный университет E-mail: physinf@phys.kubsu.ru	11
Шаталова Ольга Владимировна	305040, г. Курск, 50 лет Октября ул., 94, Юго-Западный государственный университет E-mail: shatolg@mail.ru	126
Шибзухов Заур Мухадинович	115404, Москва, ул. Гаврикова, 7/9, Институт математики и информатики Московского педагогического государственного университета E-mail: intellimath@mail.ru	132

Научное издание

Нейроинформатика, её приложения и анализ данных

Материалы XXVIII Всероссийского семинара 25 сентября – 27 сентября 2020 года

Редактор М.Ю. Сенашова
Компьютерная верстка: М.Ю. Сенашова

Подписано в печать «15» августа 2020 г. Формат 60 × 90/8. (А4)
Бумага офсетная. Печать плоская.
Усл. печ. л. 11,0. Уч.-изд. л. 11,3.
Тираж 500 экз. Заказ 15

Отпечатано в типографии ИВМ СО РАН
660036, Красноярск, Академгородок