МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
(МИНОБРНАУКИ РОССИИ)

РОССИЙСКАЯ АКАДЕМИЯ НАУК

МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЙ ИНСТИТУТ СИСТЕМНЫХ ИССЛЕДОВАНИЙ РАН

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ УНИВЕРСИТЕТ «МИФИ»

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

ТРОИЦКИЙ ИНСТИТУТ ИННОВАЦИОННЫХ И ТЕРМОЯДЕРНЫХ ИССЛЕДОВАНИЙ (ТРИНИТИ)

РОССИЙСКАЯ АССОЦИАЦИЯ НЕЙРОИНФОРМАТИКИ

РОССИЙСКАЯ АССОЦИАЦИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

XXI МЕЖДУНАРОДНАЯ НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ

НЕЙРОИНФОРМАТИКА-2019

СБОРНИК НАУЧНЫХ ТРУДОВ

ЧАСТЬ 1

- **ЛЕКЦИЯ ПО НЕЙРОИНФОРМАТИКЕ**
- **ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ**
- **ПРИКЛАДНЫЕ НЕЙРОСЕТЕВЫЕ СИСТЕМЫ**
- **НЕЙРОБИОЛОГИЯ И НЕЙРОБИОНИКА**
- **КОГНИТИВНЫЕ НАУКИ И ИНТЕРФЕЙС “МОЗГ-КОМПЬЮТЕР”**
- **НАУЧНЫЕ ТРУДЫ МОЛОДЫХ СПЕЦИАЛИСТОВ**

МОСКВА
2019

УДК 001(06)+004.032.26(06)
ББК 72я5+32.818я5
М 82

XXI МЕЖДУНАРОДНАЯ НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ "НЕЙРОИНФОРМАТИКА-2019" : сборник научных трудов**.** В 2-х частях. Часть 1. – Москва : МФТИ, 2019. 197 с. : ил.

Сборник научных трудов содержит доклады, включенные в программу XXI Международной научно-технической конференции «НЕЙРОИНФОРМАТИКА-2019», проходившей в г. Москве 7–11 октября 2019 г. Тематика конференции охватывает широкий круг вопросов: искусственный интеллект, глубокое обучение, когнитивные науки и интерфейс «мозг-компьютер», методические вопросы нейроинформатики, теории нейронных сетей, нейробиологии, модели адаптивного поведения и когнитивные исследования, нейросетевые парадигмы и приложения нейроинформатики.

УДК 001(06)+004.032.26(06)
ББК 72я5+32.818я5

ISBN 978-5-89155-322-4 (Ч. I)
ISBN 978-5-89155-321-7

ОРГАНИЗАТОРЫ КОНФЕРЕНЦИИ

- Министерство науки и высшего образования Российской Федерации (Минобрнауки России)
- Российская академия наук
- Московский физико-технический институт (национальный исследовательский университет)
- Научно-исследовательский институт системных исследований РАН (НИИСИ РАН)
- Национальный исследовательский ядерный университет «МИФИ» (НИЯУ МИФИ)
- Московский авиационный институт (национальный исследовательский университет), МАИ
- Государственный научный центр РФ Троицкий институт инновационных и термоядерных исследований (ГНЦ РФ ТРИНИТИ)
- Российская ассоциация нейроинформатики
- Российская ассоциация искусственного интеллекта

ПРЕДСЕДАТЕЛЬ КОНФЕРЕНЦИИ

Ведяхин А. А. – Первый заместитель Председателя Правления Сбербанка, Председатель Научно-координационного совета Центра компетенций по направлению «Искусственный интеллект» МФТИ

ОРГАНИЗАЦИОННЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Председатель – Гаричев С. Н., проректор по исследованиям и разработкам МФТИ, Москва
Акад. РАН Бетелин В. Б., НИИСИ РАН, Москва
Акад. РАН Евтушенко Ю. Г., ФИЦ ИУ РАН, Москва
Зам. председателя – Шумский С. А., МФТИ, Москва
Зам. председателя – Тюменцев Ю. В., МАИ, Москва
Зам. председателя – Юдин Д. А., МФТИ, Москва
Панов А. И. – МФТИ, ФИЦ ИУ РАН, Москва
Климов В. В. — НИЯУ МИФИ, Москва
Акопов Э. И. – НИИСИ РАН, Москва
Смирнитская И. А. – НИИСИ РАН, Москва
Сохова З. Б. – НИИСИ РАН, Москва
Трофимов А. Г. – НИЯУ МИФИ, Москва
Пивоваров И. О. – OpenTalks.AI
Ученый секретарь – Бесхлебнова Г. А., НИИСИ РАН, Москва

ПРОГРАММНЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Председатель – Горбань А. Н. (University of Leicester, UK & ННГУ, Россия)
Сопредседатель – чл.-корр. РАН Крыжановский Б. В. (НИИСИ РАН, Москва)
Сопредседатель – Дунин-Барковский В. Л. (МФТИ, Москва)
Abraham A. – Machine Intelligence Research Labs (MIR Labs), Washington, USA
Baidyk T. – The National Autonomous University of Mexico (UNAM)
Borisyuk R. – Plymouth University, United Kingdom
Cangelosi A. – Plymouth University, United Kingdom
Dosovitskiy A. – Albert-Ludwigs-Universität, Freiburg, Germany
Dudkin A. – United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Minsk
Golovko V. A. – Brest State Technical University, Belarus
Gorban A. – University of Leicester, Great Britain
Hayashi Y. – Meiji University, Kawasaki, Japan
Husek D. – The Institute of Computer Science of Academy of Sciences of the Czech Republic, Prague
Izhikevich E. – Braincorporation, San Diego, USA
Jankowski S. – Warsaw University of Technology, Poland
Kecman V. – Virginia Commonwealth University, USA
Kernbach S. – Cybertronica Research, Research Center of Advanced Robotics and Environmental Science, Stuttgart, Germany

Уважаемые коллеги!

Конференция НЕЙРОИНФОРМАТИКА вновь собирает исследователей, работающих по актуальным направлениям теории и приложений искусственных нейронных сетей. Как и на предыдущих наших собраниях, в этом году на конференции «НЕЙРОИНФОРМАТИКА-2019» представлены доклады по проблемам теории нейронных сетей, нейробиологии, моделям адаптивного поведения, нейросетевому моделированию объектов и систем, обработки статистических данных, временных рядов и изображений и многим другим прикладным задачам нейроинформатики.

Более 200 российских ученых и наших зарубежных коллег направили в оргкомитет конференции результаты своих исследований.

По сложившейся традиции конференцию открывают приглашенные доклады. В рамках школы-семинара участники конференции прослушают лекции известных специалистов по актуальным проблемам нейроинформатики. На мастер-классах от ведущих компаний можно познакомиться с различными приложениями искусственных нейронных сетей и инструментами для работы с ними.

Особое внимание уделяется работам студентов, аспирантов и молодых специалистов, которые примут участие в творческом конкурсе.

За прошедшие годы научно-техническая конференция «НЕЙРОИНФОРМАТИКА» сложилась как представительный и многоплановый по тематике научный форум, в работе которого принимают участие и известные ученые, и молодые специалисты, аспиранты и студенты.

Желаем всем участникам конференции плодотворной работы, активного сотрудничества и новых творческих идей!

Оргкомитет

СОДЕРЖАНИЕ

ЛЕКЦИЯ ПО НЕЙРОИНФОРМАТИКЕ

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

ПРИКЛАДНЫЕ НЕЙРОСЕТЕВЫЕ СИСТЕМЫ

НЕЙРОБИОЛОГИЯ И НЕЙРОБИОНИКА

КОГНИТИВНЫЕ НАУКИ И ИНТЕРФЕЙС «МОЗГ-КОМПЬЮТЕР»

НАУЧНЫЕ ТРУДЫ МОЛОДЫХ СПЕЦИАЛИСТОВ

ЛЕКЦИЯ ПО НЕЙРОИНФОРМАТИКЕ

Л. А. СТАНКЕВИЧ
Санкт-Петербургский институт информатики и автоматизации РАН
stankevich_lev@inbox.ru

КОГНИТИВНЫЕ ДИАЛОГОВЫЕ СИСТЕМЫ

В этой лекции обсуждаются пути развития диалоговых систем. Особое внимание отводится рассмотрению существующих архитектур диалоговых систем и возможности построения на их основе когнитивных диалоговых систем. Одно из направлений развития таких систем связано с разработкой когнитивных средств управления диалогом. Показано, что разработка таких средств актуальна для современных гуманоидных роботов, которые имеют не только антропоморфную форму, но должны осмысленно общаться с человеком на естественном языке. Обсуждаются возможности обработки естественного языка с использованием глубоких нейронных сетей. Рассматриваются варианты управления диалогом с использованием планирования и инкрементального обучения. Показано, что приемлемое для гуманоидных роботов диалоговое поведение может быть достигнуто с использованием ассоциативных средств. Рассматривается вариант когнитивной диалоговой системы гуманоидного робота.

L. A. STANKEVICH
Saint-Petersburg institute for informatics and automation of the RAS
stankevich_lev@inbox.ru

COGNITIVE DIALOG SYSTEMS

In the given lection development ways of dialog systems are discussed. The special attention is taken away to consideration existing architectures of dialog systems and possibility of building on their basis cognitive dialog systems. One of directions of development of such systems is connected with development cognitive means of dialog management. It is shown that development of such means is topical for modern humanoid robots, which have not only anthropomorphic form, but should carry out conversation with human mentality. Probabilities of natural language processing based on deep neural networks are discussed. Variants of dialog management using planning and incremental learning are considered. It is shown that

suitable for humanoid robots behavior can be achieved using associative means. Variant of cognitive dialog system for humanoid robot is considered.

Введение

Эта лекция посвящена вопросам организации диалоговых систем, которые имеют элементы когнитивности, позволяющие реализовать диалог между человеком и компьютером, подобно тому, как человек общается с человеком. Такие когнитивные диалоговые системы очень важны для роботов гуманоидного класса, поскольку главная задача таких роботов – общаться на естественном языке с человеком, не только получая от него информацию и команды, но и поддерживая осмысленный диалог с ним.

Диалоговые средства общения с компьютерными системами, позволяющие формировать задания и получать сообщения на естественном языке, относятся к человеко-машинным интерфейсам. Робот является компьютерной мехатронной системой, в которой диалог реализуется через, так называемый, интерфейс «человек-робот» (Human Robot Interface – HRI). При восприятии информации от пользователя в таких интерфейсах должны решаться прямые задачи распознавания и понимания естественно-языковых выражений, вводимых в речевой или текстовой формах. Обратные задачи, связанные с формированием смыслового содержания ответных естественно-языковых сообщений и синтеза их в речевой или текстовой форме, должны решаться, если система должна информировать о своих решениях пользователей в режиме диалога.

Диалоговые системы «человек–компьютер» стали разрабатываться с развитием вычислительной техники, чтобы дать возможность пользователям взаимодействовать с компьютерами на языке, близком к естественному. В настоящее время наиболее широкое применение нашли диалоговые системы, типа Чат-ботов (Chatbot), а также IVR-систем.

Чат-бот (также известный как smartbots, talkbot, chatterbot, interactive agent) является интеллектуальной компьютерной программой, которая реализует беседу с пользователями на определенную тему. Такие программы часто создаются, чтобы моделировать поведение человека при беседе с партнером. Некоторые из них даже способны пройти известный тест Тьюринга. Чат-боты обычно используются в диалоговых системах для разных практических целей, например, в разных потребительских сервисах или для извлечения информации. Только некоторые чат-боты используют традиционные методы обработки естественного языка, но в основном это простые системы, определяющие ключевые слова во входящих предложениях и затем отвечающие подготовленными репликами или наиболее подходящими словесными паттернами из базы данных.

Исторически первыми чат-ботами были ELIZA, разработанная в Массачусетском технологическом институте США в 1966 г. [1], и PARRY, разработанная в Стенфордском институте США в 1972 г. [2]. Из недавних разработок можно выделить A.L.I.C.E. [3], и D.U.D.E (Agence Nationale de la Recherche and CNRS, 2006). В то время как ELIZA и PARRY были использованы исключительно для симуляции типовой беседы, многие современные чат-боты используются в игровых приложениях и веб-поиске.

Как правило, чат-боты не используют сложные универсальные методы обработки ЕЯ. Обычно, при их создании используются специализированные программные продукты, позволяющие решать узко специализированные задачи. Например, A.L.I.C.E. использует язык разметки, названный AIML, который имеет набор специфических функций для разработки беседующих агентов и применяется в чат-ботах типа Alicebots.

Системы IVR (Interactive Voice Response) применяются в call-центрах и отвечают на запросы пользователей, используя предварительно записанные голосовые сообщения. Озвучивание IVR обеспечило автоматическое взаимодействие с пользователями на естественном языке. Правильно подобранное сочетание музыкального сопровождения, голоса диктора и используемой лексики создаёт благоприятное впечатление от звонка в организацию. Маршрутизация, выполняемая с помощью IVR-системы, обеспечивает правильную загрузку операторов продуктов и услуг компании.

Значительное развитие получили также системы автоматического речевого диалога для целей информационной поддержки абонентов во время путешествий и отдыха. Показательным примером такой системы является справочная система «Полетели», разработанная группой речевой информатики СПИИРАН. Эта система предназначена для получения информации о расписании авиарейсов по телефону или через Интернет. Основной дикторонезависимый модуль распознавания речи этой системы, содержащий словарь на 500 слов (наименование городов, месяцев и чисел), разработан на основе оригинальной системы распознавания русской речи SIRIUS [4], которая кратко описана далее.

В традиционных системах «человек–компьютер» диалоговый процесс обычно инициируется оператором, который начинает и поддерживает речевой диалог на основе априорной информации о теме и цели диалога, а также текущей информации о ходе диалога [5].

На первом этапе этого процесса производится распознавание речи, т. е. преобразование акустического речевого сигнала в соответствующую последовательность слов. При этом используется акустико-лексическая база данных, обеспечивающая акустический, фонологический и лексический уровни обработки речи. На самом низком акустическом уровне реализуется кодирование и параметрическое представление речевого сигнала. На фонологическом уровне выделяются простейшие, семантически значащие единицы речи

– фонемы, которые представляют собой устойчивые сочетания звуков в речи. На лексическом (морфологическом) уровне производится описание всех значащих последовательностей фонем, из которых составляются слова или части слов (морфемы).

Распознанная последовательность слов обрабатывается лингвистическим процессором. При этом используются знания о лексиконе (представлении слов текста последовательностью лексем или морфем) и грамматике, а также семантике языка (смысловые значения языковых форм), имеющиеся в базе знаний системы. Лингвистический процессор реализует синтаксический и семантический уровни обработки текста. На синтаксическом уровне с помощью правил грамматики языка производится интерпретация, т. е. строятся правильные предложения, а на семантическом – этим предложениям придается смысловое значение.

Синтез речи, необходимый для генерации ответов или вопросов системы, является обратным процессом преобразования подготовленных в системе смысловых значений в предложения, слова и речевой сигнал. При этом также используется лингвистический процессор, формирующий по смысловым значениям правильно построенные предложения, по которым дальше генерируются акустические сигналы речи для проговаривания этих предложений словами с соответствующим произношением.

Специализирует диалоговую систему для использования в определенной проблемной области прикладная часть, где главная роль принадлежит средствам управления диалогом, которые ведут обработку речи, строит контекстные интерпретации и генерирует ответы робота оператору. При этом используется семантика проблемной области и различная неязыковая информация, необходимая для эффективного управления диалогом.

В настоящее время особое внимание уделяется диалоговым системам «человек-робот», разработка которых актуальна для интеллектуальных роботов. Здесь оказались не эффективными простые подходы к пониманию естественно-языковых инструкций, основанные на простых методах ключевых слов. В процессе развития диалоговые системы роботов стали строиться с использованием методов, основанных на больших корпусах, вручную подготовленных обучающих данных для сопоставления языковых фраз с действиями робота или языком робота. Как правило, такой подход требует накапливания аннотированного корпуса фраз, что может быть очень трудоемко и учитывает только языковые вариации обучающих данных. Для повышения эффективности диалога стали использоваться когнитивные методы, в частности, обучение понятиям в реальном времени, управление диалогом с планированием и адаптацией к пользователю.

Гуманоидные роботы, имеющие антропоморфную конструкцию и способные работать вместе с людьми, должны обладать способностью вести осмысленный диалог с человеком, т. е. понимать инструкции или сообщения

неподготовленных пользователей на естественном языке и отвечать им, формируя подходящие по смыслу естественно языковые ответы или сообщения. В диалоговых системах таких роботов предполагается наличие некоторого когнитивного агента, который поддерживает беседу с пользователем, осмысленно воспринимая и синтезируя речевые или текстовые сообщения, а также может получать новые семантические понятия из информации, получаемой при обучении в процессе беседы.

Далее в этой лекции подробно рассматриваются когнитивные диалоговые системы с точки зрения их архитектуры, решаемых задач и средств реализации. Приведены примеры существующих диалоговых систем с планированием и инкрементальным обучением. Рассматривается также один из возможных вариантов когнитивной диалоговой системы гуманоидного робота, реализованный с использованием нейросетевых средств.

Архитектура когнитивных диалоговых систем

Когнитивные диалоговые системы имеют несколько обязательных компонент, решающих задачи: распознавания речи (Speech Recognition – SR), понимание естественного языка (Natural Language Understanding – NLU), управление диалогом (Dialog Management – DM), генерация естественного языка (Natural Language Generator – NLG), синтез речи (Speech Synthesis – SS). Основные задачи входят в набор обработки естественного языка (Natural Language Processing – NLP), который включает задачи понимания текстовых фраз естественного языка (NLU), а также задачу генерации фраз естественного языка (NLG). Общая схема когнитивной диалоговой системы представлена на рис. 1. Главные компоненты системы, такие как NLU, DM и NLG, объединены в рамках, так называемого, беседующего агента (Conversational Agent – CA). Такой агент имеет некоторые когнитивные свойства, связанные, прежде всего, с ментальным выводом при разборе поступивших и генерации ответных предложений, планированием и управлением беседой, а также использованием инкрементального обучения для получения новых знаний в реальном времени, улучшающих диалог.

Распознавание речи, SR, т. е. речевых сообщений, поступающих от пользователя, обеспечивает начальный этап процесса диалога – анализ речевых сигналов на основе морфологии с целью распознавания и классификации произнесенных слов в соответствии с имеющимся словарем. Обработка естественного языка, NLP, включает понимание, NLU, и генерацию, NLG, естественно-языковых сообщений. Понимание естественного языка, NLU, требует проведения синтаксического и семантического анализа для распознавания отдельных лексических единиц (слов) и понимания смыслового содержания предложений, составленных их этих единиц. Управление диалогом, DM, использует некоторую политику формирования ответа, возможно, обу-

чаемую, и решает задачу формирования смысла сообщения в ответ на распознанный смысл входного предложения. Генерация естественного языка, NLG, имеет целью формирование правильно построенных предложений в соответствии с полученным смысловым ответом. Синтез речи, SS, обеспечивает проговаривание сформированных предложений средствами генерации речи. Эта задача завершает процесс диалога.

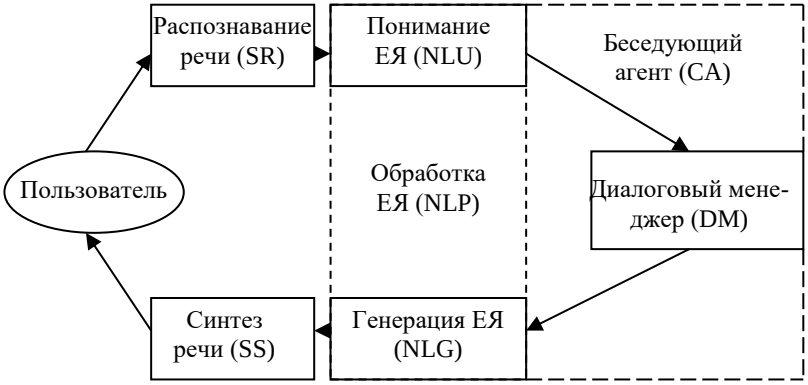


Рис. 1. Когнитивная система диалога

Далее рассмотрим более подробно обозначенные компоненты и задачи.

Распознавание речи

Традиционные системы распознавания речи последовательно решают задачи: определения границ речи, получения вектора признаков, генерации гипотез слов, собственно классификации слов. При этом качество распознавания речи зависит от: уровня шума, свойств канала передачи, размера словаря, вариативности, типа ввода речи.

В настоящее время разработаны и используются в различных приложениях, требующих речевых интерфейсов, множество вариантов систем распознавания речи, которые часто называются системами перевода речи в текст (Speech-To-Text – STT). Можно выделить следующие системы STT, способные работать с русским языком:

Speechpad – онлайн сервис, переводящий речь в текст через Google Chrome (https://speechpad.ru/);

RealSpeaker – программа для Windows, Linux, Android, Mac, переводящая речь в текст (https://transcribe.realspeaker.org),

Dragon Dictate – программа распознавания речи для мобильных устройств компании Apple.

VOCO – Windows приложение для преобразования речи в текст, разработанное в Центре речевых технологий (Россия);

SpeechKit – программа для двухэтапного распознавания речи: сначала выбираются наборы звуков, которые интерпретируются в несколько вариантов слов, а затем подключается языковая модель, которая проверяет гипотезы слов с учетом структуры языка и контекста и проводит согласование данного слова со словами, распознанными ранее (teach.yandex.ru);

Siri – голосовой ассистент компании Nuance Communications для iPhone и iPad (http:// apt.ifOrce.com);

SIRIUS – система интегрального распознавания и понимания речи, разработанная в СПИИРАН, С-Петербург, Россия.

Рассмотрим кратко принципы организации и возможности системы SIRIUS (SPIIRAS Interface for Recognition and Integral Understanding), поскольку она является одной из лучших систем, специально разработанных для полноценной обработки русской речи [6].

Система SIRIUS включает ряд программных модулей для обработки и распознавания и понимания разговорной русской речи, написанных на языках C++ и Perl для операционной системы Microsoft Window. Использованы также сторонние модули в виде исполняемых файлов.

Система SIRIUS функционирует в двух режимах: обучение и распознавание. В режиме обучения создаются модели акустических единиц речи, модель языка, а также фонематический словарь словоформ, которые используются в режиме распознавания. Для обучения акустических моделей используется вручную размеченный корпус русской речи, а для модели языка – текстовый корпус. Можно выделить следующие этапы процесса обучения: (1) обработка набора текстов и сборка текстового корпуса для разработки модели языка; (2) создание транскрипций слов их текстового корпуса и выбор наилучших транскрипций; (3) разработка стохастической модели языка; (4) обучение моделей акустических единиц речи. В режиме распознавания имеют место следующие этапы: (1) определяются границы речи; (2) производится предварительная обработка речевого сигнала; (3) речевой сигнал преобразуется в последовательность векторов признаков; (4) производится декодирование речи; (5) с использованием предварительно обученной модели языка выбираются наилучшие гипотезы слов.

Рассмотрим основные моменты, связанные с реализацией этапов обучения и распознавания речи.

Создание корпуса речи для обучения моделей акустических единиц требует записи речи многих дикторов, которым последовательно предъявляют для произнесения специально подготовленные фразы. После записи речи выполняется разметка акустического сигнала на фразы, слова и фонемы.

При обучении акустических моделей сначала производится сегментация речевого сигнала путем разделения его на короткие сегменты длительностью 10–20 мс. Затем сегменты речевого сигнала последовательно преобразуются в последовательность векторов признаков. При этом используется спектральное преобразование акустического сигнала речи, позволяющее вычислить оптимальный набор параметров этого сигнала. Цифровые отсчёты сигнала предварительно обрабатываются фильтрами, подвергаются быстрому преобразованию Фурье, проходят через набор (гребенку) перекрывающихся фильтров. Выход каждого фильтра логарифмируется и подвергается синусно/косинусному преобразованию, в результате чего получается набор кепстральных коэффициентов для каждого сегмента речевого сигнала по Мел-шкале частот.

Акустическое моделирование речи производится с использованием метода скрытых Марковских моделей (Hidden Markov Model – HMM). Этот метод основан на теории дискретных случайных цепей. Рассматривается двойной стохастический процесс: основной скрытый и наблюдаемый, позволяющий оценить основной. Скрытым является символьный процесс мозга, в течение которого давление звукового потока перерабатывается сенсорами уха некоторый код, распознаваемый мозгом. В случае применения HMM наблюдаемым процессом является снимаемый с микрофона частотный сигнал речи, который предварительно обрабатывается с целью получения наблюдаемой последовательности кепстральных коэффициентов признаков сегментов речи.

HMM представляется графом с множеством узлов-состояний и дуг-переходов между состояниями и может описывать процесс как набор последовательных состояний и переходов между ними. Каждый узел содержит вероятностное распределение состояния и вероятность, с которой данный символ наблюдается. Имея на входе последовательность наблюдаемых символов, кодирующих речь, HMM может моделировать ее вероятностными параметрами, используя Марковский процесс. Модель фонемы имеет три состояния, соответствующие началу, середине и концу фонемы. Модель слова получается соединением моделей фонем, входящих в него. Аналогично соединяются модели слов, чтобы сформировать модель фразы. Когда модель построена, может быть оценена вероятность, с которой данная последовательность генерируется из модели.

Вероятность, с которой последовательность генерируется из модели, может быть вычислена: (1) путем суммирования вероятностей всех возможных переходов и (2) процедурой forward–backward, позволяющей вычислить вероятность, с которой входная последовательность символов генерируется из модели, используя прямые и обратные переменные.

При обучении акустических моделей, основанных на HMM, производится настройка параметров модели и используется специальная процедура,

чтобы максимизировать вероятность, с которой обучающая последовательность символов генерируется из модели.

Транскрипции для собранного тестового корпуса генерируются автоматически. Вручную дополнительно создаются транскрипции для некоторых аббревиатур и слов, заимствованных их других языков. Выбор наилучших транскрипций для слов среди альтернативных вариантов осуществляется с помощью комбинированного метода. В результате работы блока транскрипций создается список фонематических представлений слов из текстового корпуса, который является фонематическим словарем системы распознавания речи.

Вероятностная модель русского языка создается на основе статистического и синтаксического анализа. Входными данными являются обучающий текстовый корпус и алфавитный словарь слов из этого текста. Сначала создается список n-грамм (групп из n синтаксически связанных слов) с частотой их появления в обучающем корпусе. Создаются n-граммы со значением n от 0 до 5. При этом выбираются только те синтаксические группы, которые были в тексте разделены другими словами. N-граммы с частотой появления меньше заданного порога, а также со словами, отсутствующими в ранее созданном словаре транскрипций, удаляются. Вероятностная модель языка в формате ARPA создается внешним модулем CMU SLM, который использует на входе список созданных n-грамм. Далее модель языка в формате ARPA трансформируется в более компактный формат SLF. Подробное описание процесса создания вероятностной модели языка можно найти в работе [6].

Процесс распознавания речи начинается с определения границ речи: начало речи определяется по моменту превышения значений энергии для речевого сигнала, порога, соответствующего значению энергии фонового шума. Затем осуществляется сегментация речевого сигнала и извлечение признаков, аналогично тому, как это делается при обучении моделей акустических единиц речи.

При распознавании отдельных слов речи создается HMM для каждого слова и реализуется процедура распознавания слов. При этом для каждого слова путем анализа речевого сигнала определяется последовательность векторов наблюдений, и затем вычисляются вероятности правдоподобия всех возможных гипотез отнесения наблюдаемого слова к некоторым словам из словаря. В результате выбирается оптимальная гипотеза наблюдаемого слова, соответствующая максимальной вероятности правдоподобия.

При распознавании слитной речи используется метод передачи маркеров, позволяющий определить прохождение возможных путей по состояниям объединенных HMM нескольких слов. В начало каждого слова из словаря ставится маркер и применяется итеративная процедура оптимизации Витерби. При этом на каждом шаге сдвигается маркер и для него вычисляется вероятностная оценка по акустической и языковой модели. После обработки

всей последовательности векторов наблюдений выбирается маркер, имеющий максимальную вероятность. Когда маркер с наибольшей вероятностью достигает конца обрабатываемого сигнала, т. е. последовательности наблюдений, то путь, по которому он проходит, становится известным, запоминается в маркере и из него считывается последовательность пройденных слов, которая и является гипотезой распознанной фразы. Если при этом было получено несколько наилучших маркеров, создается список лучших гипотез фразы, который далее обрабатывается для выбора одной наилучшей гипотезы на основе синтаксического, семантического или прагматического анализа.

В режиме распознавания речи непосредственно с микрофона пользователь произносит фразу после нажатия кнопки в окне запуска, после чего система распознает ее и выдает соответствующий ей текст.

Разработка систем распознавания речи требует большого объема акустических и текстовых данных, на основе которых создаются размеченные акустические и текстовые корпуса. Использование HMM значительно продвинуло распознавание речи в плане сложности речи, зависимости от диктора и величины словаря. Однако в последнее время на первый план выходит распознавание речи на основе нейронных сетей с глубоким обучением [7]. В продвинутых вариантах таких распознавателей речи число ошибок распознавания не превышает 5%, что, в определенном смысле, даже лучше, чем у человека. В машинном переводе также достигнуты впечатляющие результаты: нейросетевые программы делают переводы практически на уровне профессиональных переводчиков [8]. Вызывает значительные трудности обучение таких систем, поскольку требуется использовать огромное количество обучающих данных и мощные вычислительные средства.

Понимание естественного языка

Понимание естественно-языковых сообщений, которые сформированы на выходе распознавателя речи в виде цепочек, классифицированных в рамках заданного словаря слов, основано на лингвистической обработке. Стандартными задачами понимания естественного языка являются: морфологический анализ, разметка частей речи, разметка сегментов предложений, распознавание поименованных сущностей, разметка семантических ролей, синтаксический и семантический анализ.

Прежде чем рассматривать эти задачи, обсудим некоторые базовые термины в этой области. Заметим, что определения этих терминов не являются строгими с точки зрения лингвистики.

Токены – лингвистические модули, такие как слова, знаки пунктуации, числа или буквенно-цифровые индикаторы. Сегментация предложений на токены (токенизация) производится прежде, чем делать любую обработку текста. Для сегментированных языков, например английского языка, токенизацию делать относительно легко. Однако для несегментированных языков,

таких как китайский и арабский, задача является более трудной. Так, почти все символы в таких языках существуют как односимвольные слова, но могут также объединиться, чтобы сформировать мультисимвольные слова.

Корпус – набор текстов, содержащий большое число предложений. Корпуса, как правило, специализируются (аннотируются) под задачи NLP.

Тэг части речи (POS-тэг) – символ, обозначающий категорию части речи, т. е. существительное, сказуемое, прилагательное, артикль, предлог. Любое слово может быть классифицировано в одну из этих категорий и размечено POS-тегами. Символами POS-тэгов, обозначающими такие категории, являются: NN (Noun), VB (Verb), JJ (Adjective), AT (Article), PP (Preposition). Существуют общие корпуса с текстами, размеченными POS-тегами.

Дерево разбора (Parse Tree) – схематическое дерево, определяющее синтаксическую структуру предложения с использованием формальной грамматики.

Векторизация – представление текста набором кодов символов, т. е. числовым вектором, необходимое для компьютерной обработки. Самый простой способ такого представления – «мешок слов» (Bag-of-Words) или простая токенизация уникальных слов текста с последующим составлением вектора частот для предложения. Один из способов взвешивания векторов из мешка слов – TF-IDF (Term Frequency – Inverse Document Frequency) – мера, используемая для оценки важности слова в контексте документа из какого-либо набора предложений. Эти способы получили развитие в Word2vec – инструменте векторного представления слов. Такое представление основывается на контекстной близости: слова, встречающиеся в тексте рядом с похожими словами (как правило, имеющими схожий смысл), будут иметь близкие координаты своих векторов. Применяемые на практике модели Word2vec обладают осмысленной алгеброй над словами. Например, можно прибавить вектор слова «король» к вектору слова «женщина» и получить вектор, который близок к вектору слова «королева». В Word2vec существуют два основных алгоритма обучения: CBOW (Continuous Bag-of-Words) и Skip-gram [9]. CBOW предсказывает текущее слово по контексту, а Skip-gram предсказывает контекст по текущему слову. Word2vec реализован в нейросетевом варианте. В качестве входных данных сеть принимает текстовый корпус, и в процессе работы она сопоставляет каждому слову из корпуса свой вектор. В начале алгоритм проходит «окном» по текстам, составляя пары слов для обучения. Затем эти пары подаются на вход сети. После обучения последний слой этой сети отсекается, а веса скрытого слоя используются в качестве матрицы векторного представления. Фактически выходы сети являются вероятностями для входных данных (вектор слова) «увидеть» рядом какое-либо слово из набора текста.

Модель языка – обучаемая система оценки условной вероятности появления определенного слова с учетом предыдущих слов предложения. Оценивание таких условных вероятностей может быть выполнено с использованием критерия энтропии, о котором будет сказано далее. Построение таких моделей вызывает существенные трудности из-за большого объема словаря. Обычно используются n-граммные модели, позволяющие добиться низкой энтропии. Основная задача – обучить модель грамматической структуре языка, что требует больших обучающих наборов и высокой емкости модели. Критерий энтропии имеет недостаточный динамический диапазон, поскольку его численное значение в основном определяется наиболее часто повторяющимися общими фразами. Но чтобы обучить модель синтаксису, редко используемые, но правильно построенные фразы имеют не меньшее значение, чем общие фразы. Поэтому, чтобы уйти от критерия энтропии, иногда используется бинарная классификация, разделяющая корректные и некорректные фразы. Вариант модели языка на нейронной сети, описанный в работе [10], обучается по критерию ранжирования, который обеспечивает высокий ответ сети на правильные фразы и низкий – на неправильные. Модели языка, как правило, обучаются на неразмеченных данных, например, корпусе Википедии, и используются для улучшения синтаксического анализа предложений. Процесс обучения хорошей модели требует больших вычислительных средств.

Используя введенные термины, рассмотрим некоторые **общие задачи** понимания естественно языка, которые решаются в диалоговых системах.

Морфологический анализ (Stemming, Lemmatization) является начальным этапом обработки слов путем автоматического разделения их на морфемы (или stems), т. е. наименьшие лингвистические единицы, обладающие значениями. Частный случай – нахождение основной формы или леммы (lemmas) каждого слова в предложении.

Разметка частей речи (Part-Of-Speech tagging – POS-tagging) обеспечивает автоматическое назначение POS-тегов каждому слову предложения. Например, для предложения «The ball is red» после POS-тэгинга получим «The/AT ball/NN is/VB red/JJ». Эта процедура является предшествует решению более сложных задач обработки текстов, таких как семантический анализ и машинный перевод. Наилучшие POS-системы построены на классификаторах, обученных на окнах текста. В настоящее время максимальные результаты достигнуты с использованием классификаторов на методе опорных векторов (Support Vector Machine – SVM) с двунаправленным выводом (точность около 97% на слово).

Разметка сегментов предложений (Chanking) производится с целью разметки сегментов предложения с выделением групп синтаксических элементов, привязанных к существительным (NP), глаголам (VP) или предлогам (PP). Дополнительно к POS-тегам каждому слову назначается уникальный

CHUNK-тэг, часто кодируемый как begin-chunk (B-NP), inside-chunk (I-NP) или end-chunk (E-NP).

Синтаксический анализ (Parsing) позволяет сконструировать синтаксическое дерево разбора (Parse Tree) для данного предложения. Некоторые синтаксические анализаторы используют для разбора набор грамматических правил. Более эффективные анализаторы основаны на статистических моделях и могут строить деревья разбора прямо из предложений. Большинство анализаторов требуют, чтобы предложение было предварительно размечено процедурой POS-тэгинга.

Деревья разбора традиционно вычисляются с использованием и Charniak-анализаторов [11]. Современные системы часто используют дополнительные деревья разбора, оценивающие деревья разбора.

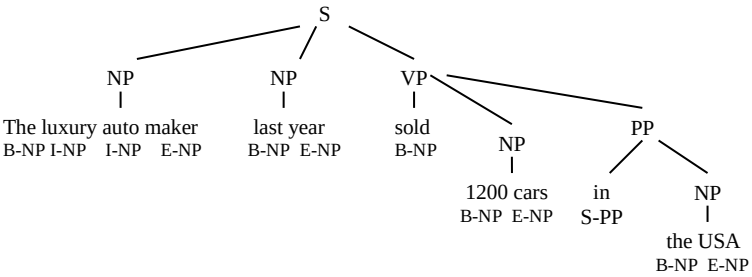


Рис. 2. Пример дерева разбора предложения

На рис. 2 приведен пример построения дерева разбора для предложения «The luxury auto maker last year sold 1200 cars in the USA» На рисунке показано оригинальное дерево разбора, полученное Charniak-анализатором. Возможно использование дополнительных деревьев разбора на нескольких уровнях. При этом уровень 0 несет информацию, ассоциированную с листьями оригинального дерева разбора. Дополнительные уровни могут быть получены неоднократным сокращением числа листьев. Для каждого уровня слова получают тэги, описывающие сегмент, ассоциированный с соответствующим листом.

Задачи синтаксического анализа часто решаются с использованием классификаторов на SVM. Использование только дерева разбора, предложенного Charniak, дает точность 74%.

Распознавание поименованных сущностей (Named Entity Recognition - NER) выполняет автоматическую разметку элементов предложений по категориям, таким как «PERSON» или «LOCATION». Каждому слову присваивается тэг суффикса или префикса, показывающий начало или середину сущности. Для решения этой задачи используются комбинации различных клас-

сификаторов. Признаками являются слова, POS-тэги, тэги суффиксов и префиксов или CHUNK-тэги. Наилучшие NER-системы достигают точности 88.8%.

Разметка семантических ролей (Semantic Role Labeling – SRL) обеспечивает автоматическое назначение SRL-тегов, определяющих семантические роли синтаксическим элементам предложения. Разработан специальный формализм ARG0-5 для назначения ролей словам, которые являются аргументами глаголов (или предикатов) в предложении [12]. Аргументы зависят от структуры глагола, и, если имеются множественные глаголы в предложении, некоторые слова могут иметь множественные тэги. Дополнительно к основным тегам ARG0-5 могут использоваться несколько модифицированных тегов, таких как ARGM-LOC (locational) и ARGM-TMP (temporal), которые действуют аналогичным образом для всех глаголов. Системы SRL решают задачу несколькими этапами: производство дерева разбора, идентификация узлов дерева разбора, которые представляют аргументы данного глагола, и, на конечном этапе, классификация этих узлов с целью вычисления соответствующих SRL-тегов. Это приводит к извлечению многочисленных базовых признаков из дерева разбора, которые используются далее в статистических моделях. Категории признаков, используемые этими системами, в частности, включают:

• части речи и синтаксические метки слов и узлов дерева;

• позиции узлов (слева или справа) по отношению к глаголу;

• синтаксический путь к глаголу в дереве разбора;

• узел в дереве разбора является частью существительного или глагола фразы;

• слово головного узла;

Этот набор признаков при необходимости может быть расширен. В общем, точность решения задачи SRL достигает 77,30% при использовании классификатора на SVM и 77,90% – в наилучшем варианте классификатора с множественными деревьями разбора.

Семантический анализ (Semantic Parsing) преобразует фразы текста в некоторую в логическую или нейросетевую семантическую форму. Логический вариант анализатора (парсера) формирует представление фразы на естественном языке в логической форме. Логический семантический анализатор определен как функция, преобразующая строки слов в семантические значения, определенные на некоторой онтологии. Он принимает последовательность токенов слов из набора, включающего набор всех токенов слов и онтологию. Онтология определяет набор констант, определяющих сущности, такие как вещи, места и люди, и предикатов, которые отображают одну или

более констант в булевы значения истины или лжи. Нейросетевые семантические анализаторы в настоящее время реализуются на глубоких нейронных сетях, обучаемых правильному отображению фраз естественного языка в семантические отношения. При этом семантический разбор непосредственно не выполняется, а семантические отношения, соответствующие введенному предложению, формируются ассоциативно, исходя из запомненных сетью примеров.

Управление диалогом

Диалог требует не только понимания естественно-языковых вопросов и сообщений, но и формирования ответов в соответствии с понятым их смыслом. Эту задачу решает диалоговый менеджер, DM.

DM поддерживает историю диалога, принимает определенную стратегию диалога, восстанавливает сохраненное в файлах или базах данных контент и выбирает лучший ответ на запрос пользователя. Решая эти задачи, менеджер диалога поддерживает процесс диалога.

Архитектуры DM развиваются в течение долгого времени в направлениях, основанных на:

- деревьях диалога;
- машинах конечных состояний;
- фреймах – система имеет несколько предварительно заполненных слотов, которые могут выполняться в любом порядке;
- деревьях планов;
- агентах, управляющих диалогом, используя методы логического или ассоциативного вывода.

При организации управления диалогом могут использоваться разные стратегии. Диалог может инициироваться системой, т. е. система сама следит за ситуацией и управляет диалогом на каждом шаге. Возможен диалог со смешанной инициативой, при котором пользователи могут изменять направление диалога, а система отрабатывает запросы пользователя, но пытается направить его, чтобы поддержать тему диалога; такая стратегия наиболее часто используется в сегодняшних системах диалога. Диалог может также инициироваться пользователем, т. е. он сам берет на себя инициативу, а система отвечает на то, что сообщает пользователь.

В диалоговой системе DM отвечает за состояние и поток беседы. Входом в DM является высказанное человеком предложение, которое обычно конвертируется в некоторое семантическое представление компонентой понимания естественного языка (ЕЯ), NLU. DM обычно поддерживает несколько переменных состояний системы, таких как история диалога, последний вопрос, на который еще не было ответа, и пр. Эти переменные состояний зависят от диалоговой системы. Выходом DM является список инструкций к исполняющей части диалоговой системы обычно в виде семантического пред-

ставления, понятного компоненту генерации ЕЯ. Это семантическое представление обычно конвертируется в ЕЯ путем использования процедуры генерации ЕЯ, NLG.

Существует много различных DM, которые по-разному управляют диалогом. Общим свойством является то, что все они поддерживают состояния диалога в контрасте с другими компонентами диалоговой системы (такими как NLU и NLG), которые только реализуют функции и не учитывают состояние. Роли DM в диалоговой системе можно разделить на следующие группы: (1) управление по входам, при котором DM выполняет зависимую от контекста обработку введенных предложений; (2) управление по выходам, при котором DM выполняет зависимую от состояния генерацию текста; (3) стратегическое управление потоком диалога, при котором DM сам решает, какое действие должно быть сделано в каждом пункте диалога; (4) тактическое управление потоком диалога, при котором DM может также принимать некоторые локальные решения, влияющие на качество беседы.

RawenClaw является представительным вариантом DM, который использован в нескольких диалоговых системах, работающих на английском языке [13]. Цель RawenClaw – непрерывно поддерживать беседу между человеком и компьютером. Этот DM выполняет две функции: интерпретацию входов пользователя по отношению к задачам проблемной области и поддержание когерентности беседы с течением времени. Для этого необходима структура задач, которая не всегда может быть явно представлена в диалоговой системе. Например, в IVR-системах структура задач неявно представлена в форме графа. В системах доступа к информации задача представляется заполненной формой, что тоже является не точным представлением. Явное представление задач, однако, необходимо для более сложных проблемных областей, например, планирования поездок. DM CMU Agenda [14] представляет один из подходов к прямому моделированию человеческих задач. Текущий вариант DM RavenClaw построен на опыте использования Agenda, особенно на обеспечении ясного отделения между поведением, определенным задачами, и более общим типом поведения, который можно назвать «стратегиями беседы». Развитие этой системы и усилия поддержки в целом фокусируются на обеспечении описания выполняемых задач; механизмов поддержания когерентности и непрерывности беседы, что обеспечивается диалоговым движком.

DM, основанные на различных принципах, используются в диалоговых системах интеллектуальных роботов. Одна из таких систем разработана, чтобы информировать и инструктировать роботов на ЕЯ [15]. Она позволяет пользователю в диалоге определять программу задач, которые должны быть запомнены и выполнены роботом, а также обеспечивает коррекцию его не-

понимания или ошибочного понимания задач. Однако понимание сообщений на ЕЯ делается путем поиска по ключевым словам и требует использования определенных слов в упорядоченной форме.

Знания робота о мире, используемые DM, могут быть модифицированы, используя семантический анализ, чтобы определить действие, пре- и постусловия окружения для этого действия и вовлеченные в действие сущности. Инструкции на ЕЯ могут определять серию действий робота. В некоторых подходах для этого используются описания пар «робот-действие» на ЕЯ, а затем строятся модели, которые отображают языковые инструкции в последовательности действий [16]. Другой подход позволяет роботу обучиться последовательности действий и лексическим единицам, которые задаются языковыми инструкциями и диалогом [17].

В других вариантах диалоговых систем используется семантический анализ, чтобы облегчить инструктирование робота на ЕЯ. Так, в одной из систем парсер обучается отображать инструкции на ЕЯ. Используется ограниченный язык и статический, вручную созданный словарь, чтобы отобразить ЕЯ в спецификации действий [18].

Обучение робота из беседы является многообещающим направлением развития диалоговых систем роботов. Так, агент диалога, имеющий базу знаний и компонент семантического понимания и способный обучаться новым выражениям во время беседы, которые направлены на инструктирование мобильного робота, описан в работе [19]. Здесь используются семантические фреймы действий и аргументы, извлеченные из предложений пользователя. Процесс автоматической генерации обучающих примеров из бесед описан также в работе [20]. Использовалась регистрация бесед, которые пользователи имели с информационной системой путешествия самолётом, чтобы обучать семантический анализатор пониманию предложений пользователя. Подход, описанный в работе [21], тоже основан на автоматическом получении примеров из беседы, но этот процесс выполняется инкрементально, т. е. в реальном времени последовательно формируются примеры по мере прохождения беседы человека с роботом. Это выполняется обучающей процедурой, основанной на λ-исчислении, включенной в интерактивную систему робота. Эта диалоговая система далее описана более подробно.

Генерация естественного языка

По результатам работы DM требуется сформировать правильно построенные фразы на ЕЯ, соответствующие смыслу ответа. Эту задачу решает модуль диалоговой системы, выполняющий генерацию ЕЯ, NLG.

Генератор ЕЯ является компонентом обработки ЕЯ, генерирующим предложения на ЕЯ по их машинному представлению в семантической фреймовой или логической форме. Лингвисты используют термин «языковая продукция», когда такие формальные презентации интерпретируются как модели для ментальных представлений. Генерация ЕЯ может рассматриваться

как обратная лингвистическая обработка, по сравнению с пониманием ЕЯ, которое выполняет прямую лингвистическую обработку речи. По существу, модули NLG должны принимать решения о том, как представить семантическую концепцию цепочкой грамматически связанных слов. Заметим, что компьютерные методы получения конечных представлений сообщений на ЕЯ отличаются от традиционных компиляторов, вследствие внутренней экспрессивности ЕЯ. Системы генерации ЕЯ разрабатываются давно, но коммерческие технологии для них только недавно стали широко использоваться.

Процесс генерации текста может быть простым, если используются заготовленные заранее фразы текста на ЕЯ, которые выбираются как ответы диалоговой системы по распознанным ключевым словам или их сочетаниям. Однако сложная система генерации ЕЯ должна включать стадии планирования и слияния информации, чтобы обеспечить генерацию текста, который выглядит естественным и не повторяющимся. Рассмотрим кратко типовые этапы генерации ЕЯ.

Определение контента. Принимается решение о том, какую информацию должен нести текст.

Структурирование документа. Информация организуется для передачи на синтезатор речи.

Агрегация. Подобные предложения сливаются, чтобы улучшить читаемость и натуральность текста.

Лексический выбор. Слова укладываются в концепты.

Генерация референтных выражений. Создаются ссылочные выражения, которые идентифицируют объекты и регионы. Эта задача связана также с решением о произношении и ударениях.

Реализация. Создание полного текста, который должен быть откорректирован в соответствии с правилами синтаксиса, морфологии и орфографии.

Альтернативным подходом к генерации ЕЯ является использование «непрерывного» (end-to-end) обучения для построения системы без реализации рассмотренных выше этапов. В этом плане разрабатываются системы генерации ЕЯ на нейронной сети с глубоким обучением, например, с использованием сети длинной и короткой памяти (LSTM), обучаемой на большом наборе входных данных и соответствующих им выходных текстов, подготовленных человеком. Этот подход можно считать самым успешным при реализации системы, которая автоматически производит текстовые описания изображений [22].

Синтез речи

Вывод акустической речи производится синтезатором речи, который восстанавливает форму речевого сигнала по его параметрам, т. е. искусственно воспроизводит человеческую речь по тексту на ЕЯ, используя правила произношения. Как правило, он может настраиваться на определенный тип речи с интонациями, чтобы усилить ее восприятие человеком. Можно считать, что

синтезатор речи является аппаратно-программной системой, способной переводить текстовые образы в речь [23].

Все способы синтеза речи можно подразделить на группы: (1) параметрический синтез; (2) компиляционный синтез; (3) синтез по правилам; (4) предметно-ориентированный синтез.

Параметрический синтез речи является конечной операцией в вокодерных системах, где речевой сигнал представляется набором небольшого числа непрерывно изменяющихся параметров. Параметрический синтез целесообразно применять в тех случаях, когда набор сообщений ограничен и изменяется не слишком часто. Достоинством такого способа является возможность записать речь для любого языка и любого диктора. Качество параметрического синтеза может быть очень высоким (в зависимости от степени сжатия информации в параметрическом представлении). Однако параметрический синтез не может применяться для произвольных заранее не заданных сообщений.

Компиляционный синтез сводится к составлению сообщения из предварительно записанного словаря исходных элементов синтеза. Размер элементов синтеза не меньше слова. Очевидно, что содержание синтезируемых сообщений фиксируется объёмом словаря. Как правило, число единиц словаря не превышает нескольких сотен слов. Основная проблема в компилятивном синтезе — объёмы памяти для хранения словаря. В связи с этим используются разнообразные методы сжатия/кодирования речевого сигнала. Компилятивный синтез имеет широкое практическое применение. В западных странах разнообразные устройства (от военных самолётов до бытовых устройств) оснащаются системами речевого ответа. В России системы речевого ответа до недавнего времени использовались в основном в области военной техники, сейчас они находят всё большее применение в повседневной жизни, например, в справочных службах операторов сотовой связи при получении информации о состоянии счета абонента.

Синтез речи по правилам обеспечивает управление всеми параметрами речевого сигнала и, таким образом, может генерировать речь по заранее неизвестному тексту. В этом случае параметры, полученные при анализе речевого сигнала, сохраняются в памяти так же, как и правила соединения звуков в слова и фразы. Синтез реализуется путём моделирования речевого тракта, применения аналоговой или цифровой техники. Причём в процессе синтеза значения параметров и правила соединения фонем вводят последовательно через определённый временной интервал, например 5–10 мс. Метод синтеза речи по печатному тексту (синтез по правилам) базируется на запрограммированном знании акустических и лингвистических ограничений и не использует непосредственно элементы человеческой речи. В системах, основанных на этом способе синтеза, выделяется два подхода. Первый подход направлен на построение модели системы, производящей речь человека, он известен

под названием артикуляторного синтеза. Второй подход — формантный синтез по правилам. Разборчивость и натуральность таких синтезаторов может быть доведена до величин, сравнимых с характеристиками естественной речи.

Синтез речи по правилам с использованием предварительно запомненных отрезков естественного языка – это разновидность синтеза речи по правилам, которая получила распространение в связи с появлением возможностей манипулирования речевым сигналом в оцифрованной форме. В зависимости от размера исходных элементов синтез может быть: микросегментный (микроволновый), аллофонический, дифонный, полуслоговый, слоговый, из единиц произвольного размера.

Обычно в качестве таких элементов используются полуслоги – сегменты, содержащие половину согласного и половину примыкающего к нему гласного. При этом можно синтезировать речь по заранее не заданному тексту, но трудно управлять интонационными характеристиками. Качество такого синтеза не соответствует качеству естественной речи, поскольку на границах сшивки дифонов часто возникают искажения. Компиляция речи из заранее записанных словоформ также не решает проблемы высококачественного синтеза произвольных сообщений, поскольку акустические и просодические (длительность и интонация) характеристики слов изменяются в зависимости от типа фразы и места слова во фразе. Это положение не меняется даже при использовании больших объёмов памяти для хранения словоформ.

Предметно-ориентированный синтез компилирует слова, записанные заранее, а также фразы для создания полных речевых сообщений. Он используется в приложениях, где многообразие текстов системы будет ограничено определенной темой/областью, например объявления об отправлении поездов и прогнозы погоды. Эта технология проста в использовании и достаточно долго применялась в коммерческих целях: её так же применяли при изготовлении электронных приборов, таких как говорящие часы и калькуляторы. Естественность звучания этих систем потенциально может быть высокой благодаря тому, что многообразие видов предложений ограничено и близко с соответствием интонацией исходных записей. А так как эти системы ограничены выбором слов и фраз в базе данных, они в дальнейшем не могут иметь широкое распространение в сферах деятельности человека, лишь потому, что способны синтезировать комбинации слов и фраз, на которые они были запрограммированы.

Первые системы синтеза речи на базе вычислительной техники стали появляться в конце 1950-х годов, а первая система, преобразующая текст в речь (Text-To-Speech – TTS), была создана в 1968 году [24].

В настоящее время имеется множество систем и инструментов для синтеза речи по тексту. Известным инструментом для синтеза русского языка

является Yandex Speech Kit, который использован для разработки чат-бота Алиса [25].

В перспективе ожидается реализация полного синтеза речи по правилам. Однако пока звучание таких систем напоминает больше всего речь роботов, которая местами труднопонимаема. Поэтому сейчас чаще используется технология разработки синтезаторов речи на обучаемых нейронных сетях.

Для гуманоидных роботов представляет интерес разработка механических синтезаторов речи, моделирующих речевую систему человека (антропоморфических моделей). Например, японские учёные из университета Waseda работают над созданием антропоморфической модели говорящего робота. Последняя их разработка (2005) — модель Waseda Talker No.5 — имеет весь набор речевых инструментов: лёгкие, гортань, мягкое нёбо, язык, зубы, губы и пр. В общей сложности все эти органы имеют 18 степеней свободы. На их страничке Anthropomorphic Talking Robot Waseda-Talker Series 2007 года можно ознакомиться с этим проектом подробнее.

Рассмотрим основные моменты, связанные с разработкой системы синтеза речи в рамках лингвистики, просодики, фонетики, акустики,

Для начала нужно провести лингвистическую нормализацию текста, т. е. развернуть все сокращения, числа и даты в текст. 50-е годы XX века должно превратиться в пятидесятые годы двадцатого века, а г. Санкт-Петербург, Большой пр. П.С. в город Санкт-Петербург, Большой проспект Петроградской Стороны. Это должно происходить так естественно, как если бы человека попросили прочитать написанное.

Следующий этап – подготовка словаря ударений. Расстановка ударений может производиться по правилам языка. Их обязательно нужно учитывать. Для русского языка в общем смысле правил расстановки ударения вообще не существует, так что необходим словарь с расставленными ударениями.

Омографы — это слова, которые совпадают в написании, но различаются в произношении. Носитель языка легко расставит ударения в сочетаниях слов «дверной замок» и «замок на горе». Более сложная задача – расстановка ударения в сочетании «ключ от замка». Полностью снять омографию без учета контекста невозможно.

Для качественного синтеза речи требуется учесть просодику, т. е. произвести выделение синтагм, расстановку пауз, а также определить тип интонации. Синтагма представляет собой относительно законченный по смыслу отрезок речи. Когда человек говорит, он обычно вставляет паузы между фразами. При синтезе речи нужно научить систему разделять текст на такие синтагмы. Выражение завершенности, вопроса и восклицания требуют разных, но простых интонаций. Сложнее выразить иронию, сомнение или воодушевление.

Синтезатор речи должен фонетически правильно произносить слова и фразы, которые записаны в виде текста. Поэтому вместо букв (графем), требуется использовать звуки (фонемы). Преобразование графемной записи в фонемную является не простой задачей, поскольку должна производиться с использованием множества правил и исключений. Нужно также решать, как при произнесении будет меняться высота основного тона и скорость произнесения в зависимости от расставленных пауз, подобранной последовательности фонем и типа выражаемой интонации.

С точки зрения акустики, важен подбор звуковых элементов. Системы синтеза оперируют так называемыми аллофонами, т. е. реализациями фонемы, зависящими от окружения. Записи из обучающих данных нарезаются на кусочки по фонемной разметке, которые образуют аллофонную базу. При произнесении каждого аллофона требуется учитывать соседние фонемы, высоту основного тона, длительность и пр. Сам процесс синтеза представляет собой подбор правильной последовательности аллофонов, наиболее подходящих в текущих условиях. Кроме того, для получившихся записей иногда нужна постобработка, какие-то специальные фильтры, делающие синтезируемую речь чуть ближе к человеческой или исправляющие какие-то дефекты.

В конечном варианте синтезатор речи должен быть проверен носителем языка, поскольку только носитель может оценить качество произносимой речи и указать на ошибки в ударениях и интонации.

Далее кратко обсудим возможности реализации синтезатора речи.

Реализацией синтезатора речи, учитывающего все сложности, связанные с тонкостями ЕЯ, является End-2-End (E2E). Эта реализация предполагает построить систему, основанную на нейронных сетях, которая на входе принимает текст, а на выходе выдает синтезированную речь.

Архитектур и подходов к E2E-синтезу довольно много. Среди основных можно выделить:

1. Tacotron (версии 1, 2);
2. DeepVoice (версии 1, 2, 3);
3. Char2Wav;
4. DCTTS;
5. WaveNet.

Эти архитектуры используются во многих синтезаторах речи. Наиболее сложными и трудно создаваемыми являются дикторонезависимые синтезаторы речи, поскольку для них требуется большой речевой корпус от разных дикторов и мощные вычислительные ресурсы для обучения. Достаточно просто создать синтезатор речи на нейронной сети с обучением по речевому корпусу, записанному от одного диктора. Однако и в этом случае при подготовке

корпуса желательно использовать профессионального диктора, который должен произносить заранее подготовленные фразы в течение многих часов. При каждом произнесении нужно выдержать все паузы, говорить без рывков и замедлений и с правильной интонацией.

Относительно простой вариант синтезатора речи реализован на глубокой сверточной нейронной сети DCTTS (Deep Convolutional Text-To-Speech) [26]. На обычном компьютере DCTTS учится примерно два дня. Заметим, что синтезатор Tacotron на таком же компьютере учился бы около двух недель.

На практике нейронная сеть DCTTS была реализована с использованием пакета TensorFlow. Для обучения сети использовался корпус на 8 часов речи, записанный профессиональным диктором. Для контроля системы кроме голосовых записей имелись точные тексты для каждой из них. В результате нейросетевой синтезатор произносил заданный текст достаточно понятно. Однако такой простой синтезатор не обеспечивает параметрическую настройку голоса и речи. Эта возможность обычно имеется только у сложных синтезаторов, таких как Tacotron.

Унифицированная нейронная сеть для обработки ЕЯ

Все описанные здесь задачи обработки ЕЯ могут быть отнесены к задачам, назначающим разные метки (тэги) словам. Традиционный подход при обработке ЕЯ: извлечение из предложения богатого набора вручную сделанных признаков, которые подаются на стандартные алгоритмы классификации, например, на базе SVM, часто с линейным ядром. Выбор признаков – полностью эмпирический процесс, базирующийся сначала на лингвистической интуиции, и затем на методе проб и ошибок. При этом для каждой задачи обработки ЕЯ необходимо проводить дополнительное исследование, чтобы подобрать наиболее эффективный набор признаков и выбор особенности. Сложные задачи, например, такие как разметка семантических ролей, требуют большого количества возможно более сложных признаков, извлеченных из дерева разбора, что может повысить вычислительную сложность. Это является существенным недостатком для крупномасштабных приложений или приложений, требующих ответа в реальном времени. Чтобы снизить вычислительные трудности, можно использовать другой подход: предварительно обработать признаки как можно меньше и затем использовать обучаемую многослойную нейронную сеть. Используя набор входных предложений, несколько глубоких слоев сети обучаются извлечению признаков, соответствующих входам. При этом используется процедура обратного распространения ошибки.

В работе [10] предложена архитектура унифицированной нейронной сети и обучающий алгоритм, который может быть применен к различным задачам

обработки ЕЯ, в частности, связанным с получением синтаксической и семантической информации при анализе ЕЯ сообщений. Эта многозадачность достигнута путем исключения из разработки специфицированных на задачу средств. Вместо того, чтобы использовать полученные вручную и тщательно оптимизированные для каждой задачи признаки входных данных, предложенная система обучается внутренним представлениям на основе большого количества немаркированных тренировочных данных. Этот подход может быть использован для построения эффективной системы анализа ЕЯ-сообщений с минимальными вычислительными требованиями.

Один из ключевых пунктов предложенной архитектуры нейронной сети – способность хорошо работать с использованием сырых или почти сырых слов. Способность обучиться хорошим представлениям слов крайне важна для решения задач обработки ЕЯ. Для эффективной обработки слова подаются на вход нейронной сети как индексы, взятые от конечного словаря D. Очевидно, простой индекс не может нести много полезной информации о слове. Однако первый слой предложенной сети отображает каждый из индексов слова в вектор признаков с использованием таблиц поиска. Исходя из особенностей задач обработки ЕЯ, соответствующее представление каждого слова дается соответствующим вектором признаков, полученным таблицей поиска, после обучения методом обратного распространения ошибки, начинающегося со случайной инициализации. Используя этот подход, можно обучить сеть очень хорошим представлениям слов из немаркированных корпусов. Предложенная сеть позволяет получить при обучении лучшие представления слов, просто инициализируя таблицу поиска слов этими представлениями (вместо случайной инициализации).

Общая многослойная архитектура, подходящая для всех описанных здесь задач обработки ЕЯ, представлена на рис. 3. Первый слой извлекает признаки каждого слова. Второй слой извлекает признаки из окна слов или от предложения в целом. Полученный набор признаков рассматривается как последовательность с локальной и глобальной структурой, т. е. она не является «сумкой слов» (bag of words). Следующие слои являются стандартными слоями нейронной сети.

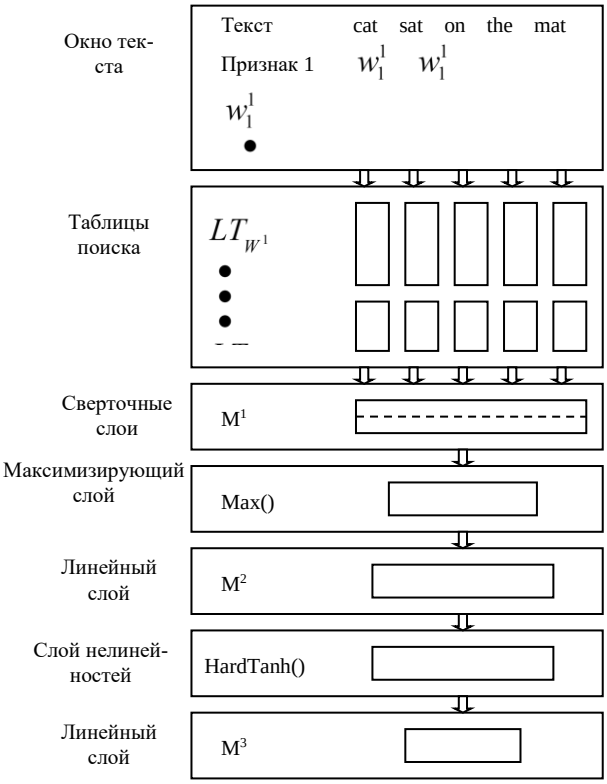


Рис. 3. Нейронная сеть для обработки окна текста

Нейронная сеть имеет L слоев с параметрами θ. Она может быть представлена композицией функций, соответствующих каждому l слою:

$$f_\theta(\cdot) = f_\theta^L(f_\theta^{L-1}(\ldots f_\theta^1(\cdot)\ldots)). \quad (1)$$

Далее опишем каждый слой сети, представленной на рис. 3.

Формально, для каждого слова $w \in D$, внутреннее представление вектора признаков размерностью d_{wrd} дается слоем таблицы поиска

$$LT_W(w) = \langle W \rangle_w^1, \quad (2)$$

где W – матрица параметров, которые получаются при обучении, $\langle W \rangle_w^1$ – w-я колонка матрицы W и d_{wrd} – размер вектора слова (гипер-параметр, выби-

раемый при настройке). Имея заданным предложение или любую последовательность из T слов $[w]_1^T$ в рамках словаря D, таблица поиска выполняет одну и ту же операцию для каждого слова последовательности, производя следующую матрицу на выходе:

$$LN_W([w]_1^T) = (\langle W \rangle_{[w]_1}^1 \langle W \rangle_{[w]_2}^1 ... \langle W \rangle_{[w]_T}^1). \quad (3)$$

Эта матрица затем подается на вход следующего слоя нейронной сети.

Заметим, что для некоторых задач обработки ЕЯ нужны не только простые признаки слов. Например, для задачи распознавания поименованных сущностей, NER, нужны признаки, которые говорят, находится ли слово в географическом справочнике или нет. В другом случае необходимо проводить некоторую базовую предварительную обработку, например, морфологический анализ с выявлением корня и окончания слова. При этом слово нужно представлять K дискретными признаками $w \in D^1 \times ... \times D^K$, где D^K – словарь для k-го признака. Каждый признак ассоциируется с таблицей поиска $LT_{W^k}(\cdot)$ с параметрами W^k и размером вектора d_{wrd}^k, который определяется пользователем. Тогда получаемый вектор признаков должен объединять векторы-столбцы признаков разного типа, т. е. $LT_{W^1,...,W^K}(w)$. Для последовательности из T слов $[w]_1^T$ слой таблиц поиска формирует матричный выход объединением таких векторов-столбцов $LT_{W^1,...,W^K}([w]_1^T)$.

Эти векторы признаков в таблице поиска эффективно получаются путем обучения для слов, входящих в словарь. Признаки, полученные при обучении слоя таблиц поиска, можно использовать как входы для последующих слоев обучаемых экстракторов признаков, которые могут представлять группы слов и затем предложений в целом.

Векторы признаков, произведенные слоем таблиц поиска, должны быть объединены в последующих слоях нейронной сети, чтобы произвести решение о назначении тэга для каждого слова в предложении. Назначение тэгов для каждого элемента в последовательностях слов переменной длины (предложение тоже является последовательностью слов) – стандартная проблема обучения машин. Далее рассмотрим два общих подхода к назначению тэгов словам: оконный подход и сверточный в случае предложения.

При оконном подходе предполагается, что тэг слова зависит главным образом от его соседних слов. При назначении тэга слову используется фиксированный размер k_{sz} (гипер-параметр) окна слов вокруг этого слова. Каждое слово в окне сначала проходит через слой таблиц поиска (3), производя матрицу признаков слова фиксированного размера $d_{wrd} \times k_{sz}$. Эта матрица может быть рассмотрена как ($d_{wrd} \times k_{sz}$)-мерный вектор, образуемый связыванием с

каждым вектором колонки. Она используется как входная для последующих слоев нейронной сети. Более формально окно признаков слова, получаемое первым слоем сети, может быть представлено как (последний символ T в верхнем регистре означает транспонирование):

$$f_\theta^1 = \left\langle LT_W([w]_1^T)\right\rangle_t^{d_{win}} = (\langle W\rangle^1_{[w]_{t-d_{win}/2}} ... \langle W\rangle^1_{[w]_t} ... \langle W\rangle_{[w]_{t+d_{win}/2}})^T . \quad (4)$$

Линейный слой. Вектор f_θ^1 фиксированного размера может быть подан на один или несколько слоев стандартной нейронной сети, которые выполняют аффинные преобразования над их входами:

$$f_\theta^l = W^l f_\theta^{l-1} + b^l , \quad (5)$$

где W^l и b^l – параметры, полученные при обучении. Имеется еще один параметр – число скрытых элементов l-слоя, который устанавливается пользователем.

Слой нелинейностей**.** Несколько линейных слоев часто формируются в стек, чередуясь с нелинейностью, чтобы извлечь сильно нелинейные признаки. Здесь в качестве нелинейности выбрана «твердая» версия гиперболического тангенса. Она дает преимущество в плане более простого вычисления по сравнению с точным гиперболическим тангенсом, оставляя при этом свойство обобщения. Соответствующий слой l применяет HardTanh-нелинейность к своему входному вектору, т. е.

$$[f_\theta^l]_i = HardTanh(x) = \{-1 \leftarrow (x < -1); x \leftarrow (-1 \le x \le 1); 1 \leftarrow (x > 1)\} \quad (6)$$

Множество выхода. Наконец, размер продукции последнего слоя L нейронной сети равен числу возможных тэгов для решаемой задачи. Каждый выход может тогда интерпретироваться как множество соответствующего тэга (задаваемое входом сети), благодаря тщательно выбранной функции стоимости, которая будет описана далее.

Сверточные слои**.** Оконный подход эффективен для большинства задач обработки ЕЯ, которые здесь обсуждаются. Однако этот подход не дает результата в задаче разметки семантических ролей, SRL, где тэг слова зависит от глагола (или, более правильно, предиката) выбранного заранее в предложении. Если глагол не попадает в окно, нельзя ожидать, что этому слову будет назначен правильный тег. В этом частном случае, назначение тэга слову требует рассмотрения целого предложения. При использовании нейронных сетей, естественным выбором решения этой проблемы становится сверточный подход.

При сверточном подходе нейронная сеть последовательно берет полное предложение, проводит его через слой таблиц поиска (3), вычисляет локальные признаки вокруг каждого слова предложения, используя сверточные

слои, комбинирует эти признаки в глобальный вектор признаков, который далее обрабатывается стандартными аффинными слоями (5). В случае разметки семантических ролей эта операция выполняется для каждого слова в предложении и для каждого глагола в предложении. Это необходимо, чтобы закодировать в архитектуре сети, какой глагол рассматривается в предложении, и какому слову требуется назначить тэг. При этом каждое слово в позиции *i* в предложении определяется двумя признаками, которые кодируют относительные расстояния $i-pos_v$ и $i-pos_w$ относительно выбранного глагола в положении pos_v, и слова, чтобы пометить в положении pos_w соответственно.

Сверточные слои, собирающие результаты таблиц поиска, можно рассматривать как обобщение оконного подхода: имея последовательность, представленную колонками в матрице таблицы поиска f_θ^{l-1} (2), к каждому окну последовательности окон применяется матрично-векторная операция

$$\left\langle f_\theta^l \right\rangle_t^1 = W^l \left\langle f_\theta^{l-1} \right\rangle_t^{d_{win}} + b^l \quad \forall t\,, \tag{7}$$

где матрица весов W^l одна и та же для всех окон t в данном предложении. Сверточные слои извлекают локальные признаки вокруг каждого окна данной последовательности. Что касается стандартных аффинных слоев (5), сверточные слои часто организуются в стек, чтобы извлечь высокоуровневые признаки. В этом случае, каждый слой должен обрабатываться нелинейностью (6), или сеть должна быть эквивалентна одному сверточному слою.

Максимизирующий слой. Размерность выхода (6) зависит от числа слов в предложении, обрабатываемом сетью. Векторы локальных признаков, извлеченные сверточными слоями, должны быть объединены, чтобы получить вектор глобальных признаков, с фиксированным размером, независимым от длины предложения. Это необходимо, чтобы применить последующие стандартные аффинные слои. Традиционные сверточные сети часто применяют операции усреднения (возможно взвешенные) или max-операции за «время» t последовательности (6). Заметим, что здесь, «время» означает только положение в предложении. Операция усреднения в нашем случае не имеет большого смысла, поскольку большинство слов в предложении не имеет никакого влияния на семантическую роль данного слова при назначении тэга. Поэтому было предложено использовать max-операцию, которая обеспечивает сети получение самых полезных локальных признаков, произведенных сверточными слоями. Имея матрицу f_θ^{l-1}, полученную сверточным слоем l − 1, максимизирующий слой l вычисляет вектор f_θ^l:

$$[f_\theta^l]_t = \max_t [f_\theta^{l-1}]_{i,t} \quad 1 \le i \le n_{hu}^{l-1} \tag{8}$$

Этот вектор глобальных признаков фиксированного размера затем обрабатывается аффинными слоями сети (4). Как и в оконном подходе, в итоге производится одно множество возможных тэгов для данной задачи.

Схемы назначения тэгов

Как было сказано, выходные слои нейронной сети вычисляют множество всех возможных тэгов для решаемой задачи. При оконном подходе эти тэги применяются к слову, расположенному в центре окна. При сверточном подходе анализируется все предложение и эти тэги применяются к слову, назначенному добавочными маркерами на входе сети.

Задача разметки частей речи, POS, выполняет назначение тэгов, определяющих синтаксическую роль каждого слова. Однако остальные три задачи назначают метки сегментам предложения. Это обычно достигается при использовании специальных схем назначения тэгов, чтобы идентифицировать границы сегмента. Существует несколько классифицирующих схем (IOB, IOE, IOBES...), но нет ясных рекомендаций по выбору наилучшей схемы. На современном уровне лучшее выполнение такой разметки иногда получается комбинированием классификаторов, обучаемых различным схемам разметки.

Для решения задач распознавания поименованных сущностей, NER, разметки сегментов предложения, CHUNK, и разметки семантических ролей, SRL, могут использоваться две базовых схемы маркировки. В предлагаемом нейросетевом варианте для решения всех указанных задач была использована самая выразительная схема IOBES. Например, в задаче CHUNK группа существительных (noun phrase) описывается четырьмя различными признаками. Признак S-NP используется, чтобы пометить группу, содержащую отдельное слово. Другие признаки B-NP, I-NP, и E-NP используются, чтобы отметить первое, промежуточные и последние слова группы. Дополнительный признак O отмечает слова, которые не являются членами сегмента.

Обучение

Все нейронные сети обучаются путем максимизации вероятности запоминания обучающих данных, используя метод стохастического градиентного спуска.

Если обозначить θ все тренировочные параметры сети при использовании θ обучающего набора T, требуется максимизировать следующую логарифмическую вероятность по отношению к θ:

$$\theta \mapsto \sum_{(x,y)\in T} \log p(y|x,\theta), \tag{9}$$

где x соответствует или окну обучаемого слова, или предложению и ассоциированным с ними признакам, а y представляет соответствующий тэг. Веро-

ятность $p(.)$ вычисляется по выходам нейронной сети. Имеют место два способа интерпретации выходов нейронной сети как вероятностей. В предлагаемом варианте каждое слово предложения считается независимым.

Вероятность уровня слов. При оконном подходе каждое слово рассматривается независимо. Имея входной пример x, сеть с параметрами θ формирует выход $[f_\theta(x)]_i$ для i тэга, относящегося к решаемой задаче. Далее для упрощения записи будем использовать обозначение $[f_\theta]_i$. Эта мера может быть интерпретирована как условная вероятность тэга $p(i|x,\theta)$ путем применения операции softmax для всех тэгов:

$$p(i|x,\theta) = \frac{e^{[f_\theta]_i}}{\sum_j e^{[f_\theta]_j}}. \quad (10)$$

Вводя логарифмическую операцию

$$\text{logaddz}_i = \log(\sum_i e^{z_i}), \quad (11)$$

можно выразить вероятность для одного обучающего примера (x,y) как

$$\log p(y|x,\theta) = [f_\theta]_y - \text{logadd}_j[f_\theta]_j. \quad (12)$$

Этот критерий обучения часто называют кросс-энтропией. Он широко применяется для задач классификации, но не является идеальным для задачи назначения тэгов, где часто имеет место корреляция между тэгом слова в предложении и тегами соседних слов. Поэтому был предложен более общий подход для нейронной сети, который учитывает зависимости между предсказанными тэгами в предложении.

Вероятность уровня предложений. В задачах разметки сегментов предложений, распознавания поименованных сущностей или разметка семантических ролей, как известно, имеются зависимости между тэгами слов в предложении. Обучение сети с использованием подхода на уровне слов бракует этот вид информации разметки. Рассмотрим схему обучения, которая учитывает структуру предложения: имея предсказанные сетью всех тэгов для всех слов предложения и имея меру связи одного тэга с другим, нужно определить правильные последовательности тэгов в процессе обучения и в то же время отсеять все другие последовательности.

Для этого введем матрицу выходов $f_\theta([x]_1^T)$ сети для предложения $[x]_1^T$, , которую упрощенно обозначим как $[f_\theta]_{i,t}$, где i – тэг t слова. Введем также меру перехода $[A]_{i,j}$ при прыжке от i к j тэгов в последующих словах и начальную меру $[A]_{i,0}$ для старта от i тэга. Поскольку эти меры переходов

должны быть получены путем обучения, как и все параметры сети θ, определим общий набор параметров как $\tilde{\theta} = \theta \cup \{[A]_{i,j} \forall i, j\}$. Мера для предложения $[x]_1^T$ вдоль последовательности тэгов $[i]_1^T$ может быть вычислена как сумма мер переходов и мер сети:

$$s([x]_1^T, [i]_1^T, \tilde{\theta}) = \sum_{t=1}^{T} ([A]_{[i]_{t-1},[i]_t} + [f_\theta]_{[i]_t,t}). \quad (13)$$

Подобно тому, как была нормализована вероятность уровня слов с использованием операции softmax (10), можно нормализовать эту меру для всех возможных последовательностей тэгов $[j]_1^T$, также используя softmax. Результат такой нормализации можно интерпретировать как условную вероятность последовательности тэгов. Применяя операцию логарифмирования, условную вероятность истинной последовательности тэгов $[y]_1^T$ можно вычислить как

$$\log p([y]_1^T \left| [x]_1^T, \tilde{\theta}) = s([x]_1^T, [y]_1^T, \tilde{\theta}) - \log \underset{\forall [j]_1^T}{adds}([x]_1^T, [j]_1^T, \tilde{\theta}). \quad (14)$$

Теперь можно максимизировать вероятность (14), используя операцию softmax (10) для всех обучающих пар $([x]_1^T, [y]_1^T)$.

В процессе вывода при назначении тэгов словам заданного предложения $[x]_1^T$ должна быть найдена лучшая последовательность тэгов, которая минимизирует меру (8), т. е. получить

$$\arg\max_{[j]_1^T} s([x]_1^T, [j]_1^T, \tilde{\theta}). \quad (15)$$

Для вывода необходимо реализовать алгоритм Витерби, т. е. выполнить рекурсию (13) и (14), но с заменой операции logadd на max, и затем проследить обратно для нахождения оптимальной последовательности тэгов на каждой операции max.

Максимизация (9) с использованием стохастического градиента производится путем итеративного выбора случайного примера (x, y) и выполнения шага по градиенту

$$\theta \leftarrow \theta + \lambda \frac{\partial \log p(y|x, \theta)}{\partial \theta}, \quad (16)$$

где λ – выбранный параметр, определяющий скорость обучения.

В итоге нейронная сеть, представленная на рис. 3, может быть составлена на основе вычисления вероятности уровня слов (12) или рекурсивного вы-

числения по выражению (14) вероятности уровня предложений (13). Аналитически сформулированное выражение с частными производной (16) может быть вычислено применением цепочечного правила дифференцирования.

Подробное описание процедур обучения представленной нейронной сети для решения рассмотренных здесь задач обработки ЕЯ можно найти в [10].

Для практического использования предложенной нейронной сети в системах обработки ЕЯ реализована программная версия этой архитектуры SENNA (Semantic/Syntactic Extraction Using a Neural Network Architecture), написанная на языке С. Эта версия проверена на задачах разметки частей речи (POS-tagging), разметки сегментов предложений (Chunking), распознавании поименованных сущностей (NER), построения дерева синтаксического разбора уровня 0 (Parse Tree level 0 – PT0) и разметки семантических ролей (SRL). При обработке текстов на английском языке достигнута хорошая точность решения этих задач: POS-tagging – 97.24%, Chunking – 94.32%, NER – 89.59%, PT0 – 92.25% и SRL – 75. 49%.

Далее кратко рассмотрим особенности использования описанной здесь архитектуры нейронной сети при решении задач векторизации, синтаксического и семантического анализа, поскольку эти задачи решаются практически во всех системах обработки естественного языка.

Преобразование слов в векторы признаков

Один из ключевых пунктов представленной нейросетевой архитектуры – ее способность хорошо решать задачи с использованием сырых или, точнее, почти сырых слов. Способность этого метода обучаться хорошим представлениям слов очень важна для предлагаемого подхода. Для эффективности слова, которые вводятся в сеть как индексы, берутся из конечного словаря. Очевидно, простой индекс не несет много полезной информации о слове. Сеть отображает каждый из этих индексов слова в вектор признаков, операцией таблицы поиска (lookup table). В соответствии с решаемой задачей, представление каждого слова дается соответствующим вектором признаков, который обучается процедурой обратного распространения ошибки, стартующей от случайной инициализации.

Для представления слов использовалась линейная версия нейронной сети, т. е. были исключены слои с нелинейностями, и оконный подход. Одного линейного слоя оказалось недостаточно: все таблицы поиска (для каждого дискретного признака своя таблица) должны быть фиксированы. Таблица поиска слова просто фиксировалась по размеру, полученному из модели языка, которая была использована. Используя нотацию, введенную для описания таблиц поиска (2), представление дискретного входа для слова w можно записать как

$$LT_{W^k}(w) = \left\langle W^k \right\rangle_w^1 = (0,...,0,1,0,...,0). \qquad (17)$$

где LT_{W^k} – таблица поиска *k*-го дискретного признака слова, а 1 внутри представления соответствует индексу слова *w*.

Эксперименты показывают, что можно обучить сеть очень хорошим представлениям слов из немаркированных корпусов.

Синтаксический анализ

В этой задаче деревья разбора традиционно вычисляются с использованием и Charniak-анализаторов [11]. Современные системы часто используют дополнительные деревья разбора, такие как *k*-вершины, оценивающие деревья разбора.

Учитывая, что узел синтаксического дерева разбора присваивает метки сегментам разбираемого предложения, был применен способ подачи этой размеченной сегментации к сети через добавочные поисковые таблицы. Каждая из этих таблиц кодирует размеченные сегменты каждого уровня дерева разбора (вверх до определенной глубины). Размеченные сегменты питают сеть по определенной схеме.

Например, к приведенному на рис. 2 дереву разбора предложения «The luxury auto maker last year sold 1200 cars in the USA» могут быть добавлены дополнительные уровни деревьев разбора. При этом уровень 0 будет нести информацию, ассоциированную с листьями оригинального дерева разбора, полученного Charniak-анализатором. Таблица поиска уровня 0 кодирует соответствующие тэги фразы для каждого слова. Дополнительные уровни 1, 2 и 3 могут быть получены сокращением числа листьев. При этом на уровне 1 меткой «O» помечаются слова, принадлежащие корневому узлу S и связанные с существительными (с тэгами NP), и остаются узлы VP и PP. На уровне 2, кроме корневого узла S, остается узел VP. На уровне 3 всем словам предложения присваивается метка «0», поскольку корневой узел S сам сокращается. Для каждого уровня слова получают тэги, описывающие сегмент, ассоциированный с соответствующим листом.

Эксперименты были выполнены с использованием модели языка, реализованной на нейронной сети, и добавочные таблицы поиска размерностью 5 для каждого уровня дерева. Они показали, что можно увеличить точность выполнения синтаксического анализа, если использовать многоуровневое дерево разбора. При пятиуровневом дереве разбора точность достигала 76%, в то время как использование только дерева разбора, предложенного Charniak (уровень 0), давала точность около 74%. Заметим, что точность синтаксического анализа важна для качественного решения задачи разметки семантических ролей, SRL.

Семантический анализ

Такой анализ имеет целью построение семантических отображений предложений, используя накопленные примеры таких отображений в процессе обучения. Описанная архитектура нейронной сети позволяет, в частности, достаточно эффективно решать задачу разметки семантических ролей, SRL. Традиционно для решения этой задачи выполняется необходимая предобработка, связанная с синтаксическим анализом текста. При решении задачи SRL в нейросетевом варианте не использовались синтаксические деревья разбора. Вместо этого, путем обучения по целевой функции, которая включает множество синтаксической информации, получались внутренние представления, несущие синтаксическую информацию о предложениях текста. Предварительно проводился синтаксический анализ с использованием многоуровневых деревьев разбора и разметки сегментов предложений, результаты которого обрабатывались таблицами поиска и подавались на вход обученной нейронной сети. В процессе синтаксического анализа использовалась модель языка на нейронной сети, обученная по нескольким корпусам английских текстов с общим размером в 130 тысяч слов.

Использование представленной архитектуры нейронной сети для решения задачи разметки семантических ролей позволило получить точность до 76%. Эта точность была достигнута с использованием предварительно полученных деревьев синтаксического разбора пяти уровней (0–4). Использование в качестве предварительной обработки только разметки сегментов предложений (Chunking) привело к снижению точности до 74%.

Когнитивный диалоговый агент с планированием

Основу когнитивной диалоговой системы составляет диалоговый менеджер, который управляет процессом разбора фраз пользователя и формированием сообщений. Далее описывается структура диалогового менеджмента ReinForest, ориентированная на создание когнитивной диалоговой системы, независимой от пользовательской области [27]. Этот диалоговый менеджер формализует существующую структуру диалогового менеджмента RavenClaw [13], основанную на использовании полумарковского процесса принятия решений (Semi-Markov Decision Process – SMDP), который является базовым в иерархическом обучении с подкреплением (Hierarchical Reinforcement Learning – HRL). HRL и SMDP были кратко описаны во второй части книги, поскольку на них базируется обучение когнитивных структур и агентов. Предложенная структура также включает зависимую от области онтологию для разработчиков, чтобы быстро закодировать знание из определенной области. Это позволяет осуществить быстрое расширение системы на

новые области и обеспечивает возможность многократного использования политики диалога.

Диалоговый агент, построенный на основе диалогового менеджера, названного ReinForest, представлен на рис. 4. Входом в ReinForest является семантический фрейм, получаемый на выходе внешней подсистемы понимания естественного языка, NLU, а выходом – список диалоговых действий и пар смысловых значений для подсистемы генератора фраз на естественном языке, NLG.

Диалоговый менеджер ReinForest имеет две части: Онтологию Знаний (Knowledge Ontology) и Движок Диалога (Dialog Engine). Онтология Знаний – зависимый от области граф знания, где разработчики могут быстро закодировать знания проблемной области и отношения между различными понятиями. В то же время, Движок Диалога – независимый от проблемной области механизм выполнения, который производит ответ системы, соответствующий текущему состоянию диалога. Последовательно работающими компонентами Движка Диалога являются: Выполнение Иерархической Политики (HPE – Hierarchical Policy Execution), Коррекция Убеждений (BE – Belief Update), Преобразование Дерева (TT – Tree Transformation) и Обработка Ошибок (EH – Error Handling).

Семантический фрейм, формируемый подсистемой NLU, содержит оригинальное предложение на естественном языке и три аннотации: сущность, намерение и область (entities, intent, domain). Например, семантический фрейм предложения пользователя: «Recommend a restaurant for me in Pittsburgh», поступающего на вход ReinForest, имеет следующий вид:

Domain {Restaurant: 0.95; Hotel: 0.05}

Intent {Request: 0.9; Inform: 0.1; Confirm: 0.0}

Entities {Type: Location; Value: Pittsburgh}.

Имея на входе аннотированное с помощью подсистемы NLU предложение пользователя, ReinForest будет корректировать внутреннее состояние диалога и генерировать следующий ответ системы в формате списка диалоговых действий и контента, т. е. asys = $[(DA_0,v_0),...(DA_k,v_k)]$.

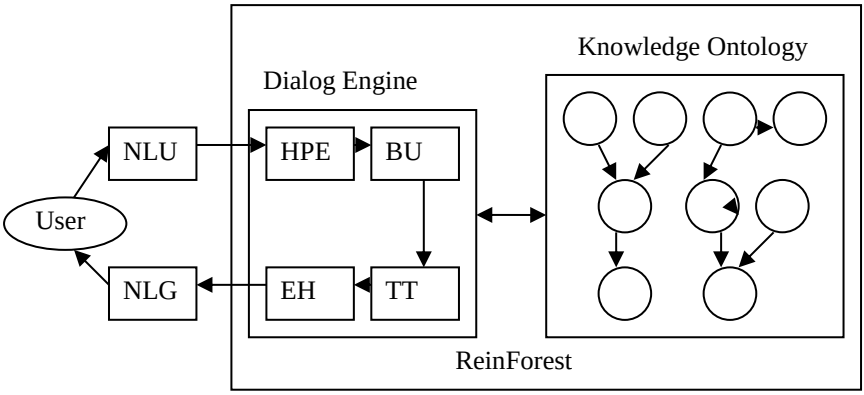


Рис. 4. Диалоговый агент [27]

Подсистема NLG является ответственной за преобразование asys в естественно-языковую форму и выдачу ответа пользователю. Для приведенного ранее примера ReinForest преобразует исходное предложение пользователя в форму

asys = [(CONFIRM, value= Pittsburgh),(ASK, value = foodtype)].

Далее подсистема NLG использует эту форму, чтобы сгенерировать следующий ответ:

«I believe you said Pittsburgh. What kind food do you want?»

Определим более формально главные компоненты Онтологии Знаний диалогового менеджера ReinForest.

Онтология знаний

Онтология знаний (Knowledge Ontology) может рассматриваться как источник знаний для движка диалога ReinForest. Основная единица онтологии – **концепт**. Внутренняя структура концепта обрисована в общих чертах на рис. 5 и формально определена как:

1. Отображение атрибута (AtributeMap): по существу, смешанное отображение, которое отображает ключевую строку в атрибут (определенный ниже).

2. Набор подписанных сущностей и областей (Subscribed Entities): когда входное предложение пользователь содержит подписанные сущности или области, движок диалога вызовет функции обработки, чтобы выполнить коррекцию концепта.

3. Набор концептов зависимостей: концепт может зависеть от ряда других концептов, прежде чем может быть получено его значение. В настоящее время ReinForest не позволяет использовать циклические графы, поэтому онтология знаний формируется в виде прямого ациклического графа (Direct Acyclic Graph – DAG). Практически зависимости кодируют знания проблемной области. Например, чтобы знать о погоде, агент сначала должен спросить дату и местоположение.

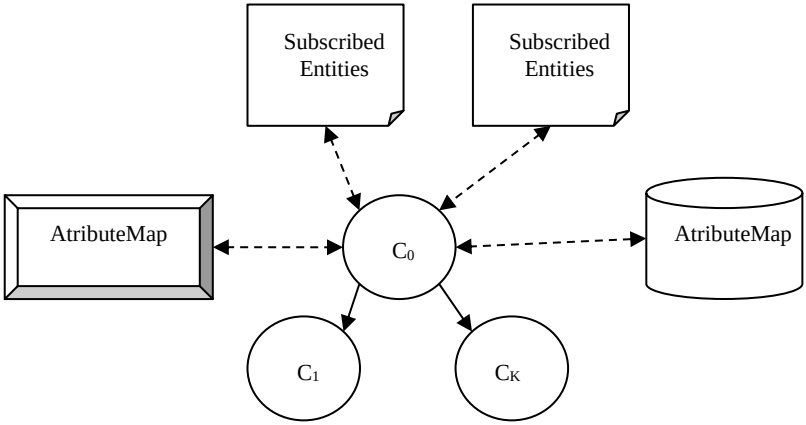


Рис. 5. Пример структуры концепта [27]

Атрибут является базовой единицей памяти концепции. Он является классом объекта, который содержит:

1. Ключ: уникальный ID атрибута.
2. Значение: сырое значение входа пользователя.
3. Нормализованное значение: значение, которое нормализуется по определенному правилу, например, сегодня 4.11.2018.
4. Счет: счет указывает уровень уверенности.

Введение определения концепта, ReinForest позволяет разработчикам строить знания проблемной области в форме DAG. ReinForest также вводит идею пула концептов, чтобы создавать группы концептов. По существу, пул концептов – коллекция концептов, которые имеют некоторые общие свойства.

По умолчанию имеется два пула концептов: пул концептов агента и пул концептов пользователя. Пул концептов агента состоит из концептов, которые агент знает, например, имя агента, информация о ресторане пр. С другой стороны, пул концептов пользователя содержит концепты, которые знают только пользователи, например, дата, которую пользователи запрашивают,

имена пользователей. Например, в простом диалоговом менеджере пул концептов агента включает ресторан и имя, поскольку агент знает свое имя и название ресторана. Чтобы получить информацию о ресторане, он должен запросить от пользователей 3 концепта: цену, локацию и тип. Поэтому концепт ресторана из пула концептов агента зависит от трех концептов из пула пользователя.

Движок диалога

Как показано на рис. 4, движок диалога ReinForest, который является независимым от проблемной области механизмом выполнения диалога, состоит из четырех главных компонентов: иерархического выполнения политики, HPE, коррекции уверенности, BU, преобразования дерева, TT, и обработки ошибок, EH.

Ниже приведен псевдокод алгоритма главного цикла выполнения процедур внутри ReinForest. Движок диалога устанавливает связь с онтологией знаний через состояние диалога, s, которое собирает всю информацию о продолжающихся диалогах. Пулы концептов агента/пользователя – оба части s состояния диалога.

```
begin ReinForest Main Loop
  while dialog.end() ≠ True do
    while dialog stack.top().type6= PRIMITIVE do
      execute(dialog_stack.top_agent)
    end while
    if user has input then do
       belief_update()
       tree_transformation()
       error_handle()
    end if
  end while
```

Далее формально определим состояние диалога, диалоговую политику и каждый из указанных четырех модулей в движке диалога.

Состояние диалога содержит указатель на онтологию знаний, а также экстраинформацию о диалоге. ReinForest содержит простые переменные, которые полезны для принятия решения. Некоторые переменные состояния включают, например, число изменений предложений, аннотацию предыдущего предложения, предыдущую проблемную область и т. д.

Иерархическое выполнение политики, HPE, предполагает использование иерархического обучения с подкреплением (Hierarchical Reinforcement

Learning – HRL). ReinForest формализует управление диалогом, основанное на планах (plan-based dialog management), на языке HRL [13]. Это открыло возможность применения хорошо отлаженных алгоритмов HRL для оптимизации действий основанного на планах диалогового менеджера. Вспомним нотацию HRL и определим задачу построения дерева диалога, используя этот язык.

HRL в основе имеет Марковский процесс принятия решений (Markov Decision Process – MDP), который описывается как:

1. S – пространство состояний диалога.

2. A – набор примитивных действий.

3. $P(s_0|s,a)$ – вероятность перехода в состояние s и выполнения примитивного действия a в этом состоянии.

4. $R(s_0|s,a)$ – функция подкрепления, определенная на S и A.

MDP, M, может быть декомпозирован на набор подзадач $O = \{O_1,O_2,..O_k\}$, где O_0 является корневой подзадачей и решение подзадачи O_0 решает M. Подзадача является полумарковским процессом принятия решений (Semi-Markov Decision Process – SMDP), который характеризуется двумя добавочными переменными по сравнению с MDP:

1. $\beta_i(s)$ – терминальный предикат подзадачи O_i , которая разделяет S на набор активных состояний S_i и набор терминальных состояний T_i. Если O_i входит в состояние в T_i, O_i , и его потомки немедленно выходят, т. е. $\beta_i(s)$ =1, если $s \in T_i$, иначе $\beta_i(s)$ =0.

2. U_i – не пустой набор действий, которые могут выполняться в задаче O_i. Эти действия могут быть или примитивными действиями из A, или другими подзадачами O_j, где $i \neq j$. Можно считать U_i детской задачей подзадачи O_i.

По существу, иерархическая политика для M есть π и является просто набором политик для каждой подзадачи, т. е. $\pi = \{\pi_0, \pi_1, ..., \pi_n\}$. Очевидно, что действительная иерархическая политика формируется как DAG, корни которого идут из O_0 и терминальные листья являются примитивные действия, принадлежащие набору A.

Диалоговая политика ReinForest формирует диалоговое дерево задач. Каждый узел дерева является подзадачей одного из трех типов: диалоговое агентство, агентство выбора диалога, диалоговый агент.

• Диалоговое агентство: подзадача $O_i \in O$ с фиксированной политикой, которая выполняется ее детьми слева направо.

• Агентство выбора диалога: подзадача $O_i \in O$ с обучаемой политикой, которая выбирает следующую детскую задачу, основываясь на контексте.

• Диалоговый агент: примитивное действие $a \in A$, которое действительно представляет действия пользователей.

При отображении диалогового дерева задачи используются специальные нотации для представления трех указанных базисных подзадач типов узлов.

Коррекция уверенности (BU) имеет место, когда есть новый вход от пользователей. Этот компонент будет сначала корректировать общие состояния диалога, такие как инкрементальное увеличение счетчика изменений состояний диалога. Затем он проходит циклически через все концепты онтологии знаний и проверяет, соответствуют ли аннотации в новом входе с какой-нибудь подписанной областью/намерением/сущностью в каждом концепте. Если найдено положительное сопоставление, новые значения запоминаются в карте атрибута концепта.

Преобразование дерева, TT, является ключевым в ReinForest, чтобы поддержать смешано-инициативные беседы во многих проблемных областях. Преобразование включает два шага: генерацию дерева кандидата и выбор дерева кандидата. Генерация деревьев-кандидатов производится путем сканирования всех обновлений, сделанных коррекцией уверенности, и формируется список деревьев кандидатов, которые могут быть помещены в стек диалога. Обычно кандидат генерируется, когда пользователь явно запрашивает информацию, которая обрабатывается в отличной от текущей проблемной области. Выбор кандидата осуществляется из числа деревьев-кандидатов агентством выбора диалога, которое выбирает одно из деревьев и помещает его в стек диалога.

Обработка ошибок, EH, предполагает обработку неправильно понятых ошибок и обработку не понятых ошибок. Обработка ошибочно понятых ошибок используется, чтобы провести явные или неявные подтверждения о концептах онтологии знаний. Специфично движок диалога проходит по циклу через все концепты в пуле концептов пользователя и выбирает концепты, которые обновлены, но еще не готовы для обработки неправильно понятых ошибок. Подзадача неправильного понимания будет тогда помещена в стек. Подзадача неправильного понимания выберет встроенную стратегию обработки таких ошибок, чтобы подтвердить каждый концепт. Текущее выполнение поддерживает два типа стратегий: неявное и явное подтверждение. С другой стороны, обработка непонятностей активизируется, когда есть пользовательский вход, но никакое обновление или преобразование дерева не дает успешного результата. ReinForest реализует широкий диапазон стратегии обработки непонятностей, в пределах от простого: «Вы можете повторить это?» до ответа, исходя из внешнего Чат-бота.

Дополнительными средствами являются: дерево информации, дерево запросов, дерево ошибочного понимания и дерево непонимания. Дерево информации используется для информирования концепта из пула концептов агента. Корень дерева информации – агентство диалога, оно выполняет свои детские задачи слева направо. Дерево информации строится рекурсивно и

позволяет извлечь все зависимые концепты. Дерево Запроса – простое дерево, с корнем в агентстве выбора диалога. Его дети содержат каждое измерение концепта. Так как корень – агентство выбора, порядок выполнения зависит от специфики политики. Дерево ошибочного понимания используется для простого выполнения обработки ошибок неправильного понимания. Корень имеет список агентств выбора диалога, каждое из которых отвечает за подтверждение специфического концепта. Каждое агентство выбирает стратегию обработки неправильного понимания для каждого концепта. Дерево непонимания реализуется как агентство выбора диалога со списком стратегий обработки непониманий как детских задач. Это дерево используется, когда новый пользовательский вход не может быть понят полностью. Текущий вариант ReinForest реализует четыре стратегии обработки непониманий: Повторить вопрос, Просить перефразировать, Зарегистрировать непонимание и Чат-бот.

Представленная простая структура диалогового менеджера ReinForest пригодна для быстрой разработки менеджера диалога во многих проблемных областях. Этому способствует специальная организация ReinForest: движок диалога является проблемно независимым, а настройка диалога на определенную проблемную область обеспечивается проблемно зависимой онтологией знаний.

Когнитивный диалоговый агент с инкрементальным обучением

Когнитивный диалоговый агент, описанный в работе [21], интегрирует семантический парсер, производящий представления фраз пользователей в логическую семантическую форму, и диалоговый менеджер, который поддерживает состояние убеждения для цели пользователя. Агент обучается инкрементально, т. е. стартует с малым числом обучающих примеров для парсера и индуцирует дополнительные примеры в процессе естественно языковых диалогов с разными пользователями. Когда агент понимает цель пользователя, он находит соответствующее представление в семантической форме, но при непонятных фразах, произнесенных в процессе беседы, он формирует новые обучающие примеры для семантического парсера. Это позволяет агенту инкрементально обучаться новым семантическим понятиям, связанным с неизвестными до этого словами. Такой подход является более робастным, чем поиск по ключевым словам, и требует малого количества начальных данных. Кроме того, он может быть использован в любом контексте, где роботам заданы высокоуровневые цели на естественном языке. В результате робот с таким когнитивным агентом будет обладать робастной способностью извлечения лексических единиц информации из любых данных.

Далее рассмотрим подробнее структуру когнитивного диалогового агента и возможности построения обучаемого семантического парсера, а

также диалогового менеджера, управляющего процессами понимания естественно-языковых сообщений и синтеза ответных сообщений.

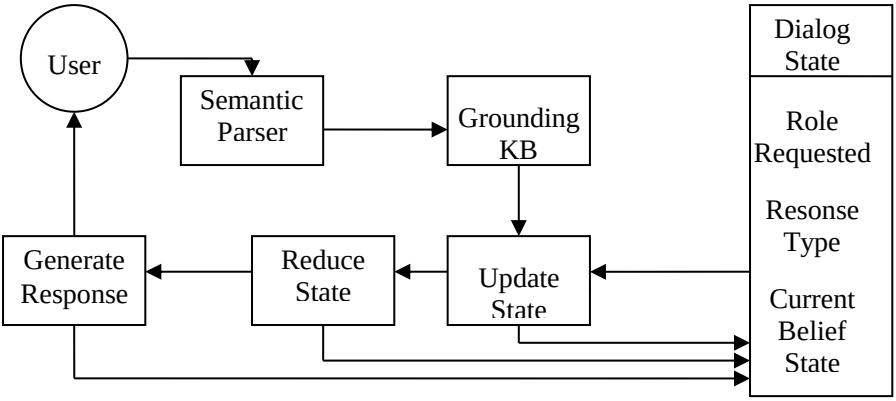


Рис. 6. Структура когнитивного диалогового агента

Структура агента представлена на рис. 6. Пользователь, User, сначала дает команду агенту, а затем начинается диалог, в котором пользователь вводит инструкции на естественном языке, которые обрабатываются семантическим парсером, Semantic parser. Парсер активизирует базу знаний, Grounding KB, в результате чего выбирается нужная инструкция на внутреннем языке системы. Агент может спрашивать разъяснения у пользователя через модуль генерации ответа, Generate Response, если он не может понять эти инструкции. При этом модули Update и Reduce State могут корректировать состояние диалога, Dialog State. Агент поддерживает текущее состояние уверенности, Current Belief State, о цели пользователя. Когда он находится в этом состоянии, т. е. понимает цель, диалог заканчивается и цель выполняется роботом. В состоянии диалога состояния уверенности, Belief State, учитывается запрашиваемая роль, Role Requested, и тип ответа, Response Type.

Обучаемый логический семантический парсер

Семантический парсер производит представление фразы пользователя в некоторую семантическую форму. Один из вариантов его реализации – формирование представления в логической форме того, что сказал пользователь на естественном языке. При этом используется λ -исчисление и комбинаторная категориальная грамматика. λ -исчисление является формализмом для представления значений в лексических единицах. Комбинаторная категориальная грамматика (Combinatory Categorial Grammar – CCG) [28] позволяет определять синтаксические категории входных последовательностей лексических

единиц и иерархически комбинирует их в синтаксическое дерево. Эта грамматика популярна и используется, как в старых, так и в текущих моделях синтаксического парсинга.

Логический семантический парсер определен как функция, преобразующая строки слов в семантические значения, определенные на некоторой онтологии. Формально семантический парсер P можно представить выражением: P(T) ×LO → S0 принимает последовательность токенов слов $x \in P(T)$ из набора T, включающего набор всех токенов слов и лексикон LO для онтологии O, и получает на выходе семантический разбор (parse) $s \in SO$ из набора всех семантических разборов в онтологии O. Онтология O определяет набор констант и предикатов. Константы определяют сущности, такие как единицы, места и люди. Предикаты отображают одну или более констант в булевы значения истины или лжи (например, red есть предикат, который говорит, что аргумент константы единицы есть красный по цвету). Этот лексикон L включает структуру данных с информацией, отдельные токены слов соотносятся с онтологией. Например, токен alice соответствует онтологической константе alice, и он обладает собственным маркером и предикатом (рис. 7). Семантический разбор является представлением значения в терминах онтологических предикатов и λ -выражений, которые позволяют абстрагировать значения на наборе онтологических констант и предикатов. Так, значение «go to alice's office» может быть представлено как:

$$go(the(\lambda x.(office(x) \wedge owns(alice, x))))$$

В этом случае будет выбрана некоторая уникальная константа x, которая означает office и принадлежит alice, иллюстрирующая пример, что константа есть аргумент для команды go. Обучение семантического парсера выполнению этих трансляций не является тривиальным, поскольку использует статистический парсинг с композиционной семантикой через CCG.

CCG добавляет синтаксическую категорию к каждому токену входной последовательности ~x и затем иерархически комбинирует эти категории, чтобы сформировать синтаксическое дерево. Эти синтаксические категории типично являются малым набором базисных категорий, подобно N (noun) и NP (noun phrase), вместе с объединенными сущностями, такими как PP/NP (prepositional phrase). В любом композиционном семантическом фреймвоке входной лексикон L содержит сущности, отображающие токены слов в семантические значения и синтаксические категории CCG. Семантические категории могут быть объединены в соответствии с иерархией, установленной через синтаксическое дерево CCG, чтобы сформировать семантический разбор, который объединяет последовательность токенов сущностей. Рис. 7 демонстрирует это для выражения Alice'soffice.

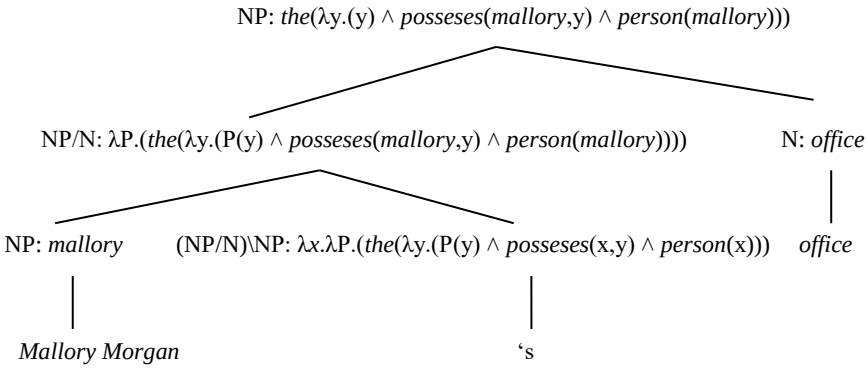


Рис. 7. Разбор выражения *Alice’s offic* [21]

Сначала парсер инициализируется малым лексическим набором, связывающим лексические единицы с синтаксическими CCG-категориями и λ -выражения, а затем тренируется на малом наборе супервизорно полученных пар «предложение-логическая форма». Имея логическое лямбда-выражение, агент может установить некоторые переменные путем запросов к базе знаний фактов о среде. Так, имея выражение «Mallory Morgan’s office», агент может установить, что y = 3508 удовлетворяет выражению, поскольку офис 3508 принадлежит Mallory. Если имеется много удовлетворяющих выражению объектов, все они возвращаются, но при этом уменьшается уверенность агента в корректности вывода.

Состояние уверенности агента относительно цели пользователя имеет три компонента: action, patient, recipient. Каждый компонент – гистограмма уверенностей по возможным назначениям. Агент поддерживает два действия: ходьба и перенос единиц, таким образом, состояние уверенности для действия – два значения уверенности [0,1]. Recipient и patient могут иметь ценности по месту сущностей (люди, комнаты, единицы) в основе знания так же как пустой ценности 0. Все уверенности калибруются к нулю, когда начинается новая беседа.

Коррекция состояния уверенности. Множественные гипотезы значений могут быть генерированы из предложений пользователя. Рассмотрим такой пример:

выражение go to the office

логическая форма $action(walk) \wedge recipient(walk, the(\lambda y.(office(y))))$.

Для n офисов эта логическая форма имеет n оснований, производящих различные значения. Агент может быть уверен, что ходьба – задача, но ее уверенность в n значениях для recipient ослабленная. Можно использовать простое обновление уверенности, основанное на k гипотезах, произведенных, чтобы отследить уверенность агента в его понимании каждого компонента запроса. Для утверждения, инициируемого пользователем, агент обновляет все компоненты состояния уверенности. Для каждой гипотезы кандидата $H_{i,c}$ с $0 \le i < k, c \in \{\text{action,patient,recipient}\}$, агент корректирует так:

$$conf(c = H_{i,c}) \leftarrow conf(c = H_{i,c})(1 - \frac{\alpha}{k}) + \frac{\alpha}{k}.$$

Здесь $0 < \alpha < 1$ – порог уверенности, выше которого кандидат принимается без дальнейшего разъяснения. Уверенность в неупомянутых аргументах разрушается, чтобы снять предыдущие недоразумения, когда пользователя попросили перефразировать цель. Для A_c, набор всех кандидатов компонента c, $\overline{A}_c \setminus \cup_i \{H_{i,c}\}$ не обозначен. Для каждой $\overline{H}_{j,c} \in \overline{A}_c$ агент корректирует так: $conf(c = \overline{H}_{j,c}) \leftarrow \gamma conf(c = \overline{H}_{j,c})$, где $0 \le \gamma \le 1$ является параметром разрушения. Ответы, инициированные системой, ассоциируются с запрошенным компонентом. Они могут принять форму соглашений или подсказок для компонентов. Для предыдущего утверждение пользователя обновит уверенность всех упомянутых значений к 1. Для последующего положительные и отрицательные обновления, описанные выше, воздействуют только на требуемый компонент.

Формирование ответа. Агент использует статическую политику диалога π, действующую на дискретном наборе состояний, составленных из кортежей action, patient, recipient вместе с ролью, которая должна быть разъяснена. Непрерывная уверенность агента S уменьшается до дискретного состояния S', путем рассмотрения главных аргументов кандидата T_c для каждого компонента c: $T_c = \arg\max_{t \in A_c} (conf(c = t))$.

Каждый компонент c в S' селектирован путем выбора T_c или «unknown» с вероятностью conf (c = T_c). Компонент c с минимумом уверенности выбран как роль, которую следует запросить. Если выбран «unknown» для каждого компонента, ролью, которая запрошена, является «all». Если «unknown» не выбран ни для какого компонента, ролью, которая запрошена, является «confirmation». Если каждое из значений уверенности, инспектированных во время этого процесса, превышает α, беседа заканчивается. Во всех экспериментах использовались параметры α = 0.95, γ = 0.5.

Обучение из беседы

Семантический парсер использует основанную на шаблоне лексическую процедуру генерации (GENLEX), описанную подробно в [29], чтобы добавить новые лексические единицы, соответствующие CCG, и логические формы, произведенные из существующих входных записей. Для каждого предложения, спаренного с логической формой и полученной во время обучения, GENLEX производит лексические единицы (n-граммы предложения) и спаривает их со всеми комбинациями n-грамм логических констант. Пары, которые приводят к успешным разборам, добавляются к словарю парсера. Обучение также повторно взвешивает параметры парсера, влияя на вероятности отдельных назначений лексических единиц логическим формам. Так, если парсер воспринимает неправильную пару, первоначально будут ошибочные понимания пользователя, но далее пара будет дискредитирована и удалена, поскольку появятся правильные контексты.

Агент индуцирует обучающие примеры для парсера из бесед с пользователями, чтобы обучиться новым лексическим единицам. Он использует заключения при диалоге и явные соглашения пользователей как супервизоров. Семантический парсер, представленный ниже, не знает неправильное написание слов calendar, planner или числа 5 (место, где лежит calendar). Когда пользователь просит, чтобы item in slot 5 была бы доставлена, он только уверенно обнаруживает действие, bring, и цели пользователя. Recipient Daniel разъясняется вопросом, инициированным системой. Когда агент спрашивает о согласовании действия, а пользователь не отрицает это, увеличивается уверенность агента. В процессе разъяснения для patient, пользователь неявно обеспечивает свидетельство, что calendar и planner – то же самое. Когда две или более фразы используются в том же самом поддиалоге, чтобы разъяснить аргумент, возможная отобранная логическая форма соединяется с более ранними поверхностными формами для того, чтобы переобучиться. Далее представлен пример диалога с роботом.

ROBOT *How can I help*?
YOU *please bring the item in slot 5 to Daniel*
ROBOT *To whom should I bring something*?
YOU *Daniel*
ROBOT *What action did you want me to take involving Daniel*?
YOU 5
ROBOT *What should I bring to Daniel*?
YOU *a calendar*
ROBOT *I am sorry, but I could not pinpoint what you meant by that*
ROBOT *What should I bring to Daniel*?
YOU *a day planer*
ROBOT *I am sorry, but I could not pinpoint what you meant by that*
ROBOT *What should I bring to Daniel*?
YOU *a calendar*
ROBOT *You want me to bring calendar to Daniel*?
YOU *yes*

Инициированные пользователем ответы генерируют подобные выравнивания. Одна из бесед пользователя началась с запроса please report to room 3418, который агент не мог разобрать из-за нового слова report. Агент понял перефразированный запрос go to room 3418, и прежнее предложение было соединено с логической формой этого последнего запроса для обучения. Когда процедура GENLEX исследовала возможные семантические значения для report, она нашла правильный парсинг со значением go, $S / PP : \lambda P.(action(walk) \wedge P(walk))$ и добавила его к лексикону парсера. Это значение говорит, что «report» должно сопровождаться препозициональной фразой, определяющей цель для действия типа походки.

Агент был использован для офисной среды и для общения с мобильным роботом Segway, действующим на этаже здания информатики университета.

Когнитивная диалоговая система гуманоидного робота

Разработка речевых диалоговых систем может проводиться с использованием программных средств, таких как GATE, NLTK, Deeppavlov и др.

GATE (General Architecture for Text Engineering) – инфраструктура для разработки и развития программных компонентов обработки текстов на естественном языке с открытым кодом. Разрабатывается и развивается более 20 лет в университете Шеффилда, Великобритания [30]. GATE основан на языке Java и обеспечивают, помимо структуры, общую архитектуру, которая описывает, как языковые компоненты обработки соединяются друг с другом,

а также графическую среду. GATE свободно доступен и прежде всего используется для раскопки текста и экстракции информации.

NLTK (Natural Language Toolkit) – набор программных инструментов для обработки естественного языка на языке Python [31]. NLTK является набором модулей и текстовых корпусов, реализованных как открытый лицензированный источник, который позволяет изучать и проводить исследования в области NLP. Самое важное преимущество использования NLTK состоит в том, что он полностью самостоятелен. Кроме того, что он обеспечивает удобные функции и программы, которые могут использоваться как стандартные блоки для общих задач NLP, он также содержит сырые и предварительно обработанные версии стандартных корпусов, используемых в литературе по NLP. Полное описание NLTK приведено на сайте этого проекта [32].

Deep Pavlov – проект по искусственному интеллекту и глубокому обучению, направленный на разработку и развитие программной платформы для создания естественно-языковых диалоговых систем с открытым кодом на языке Python [33]. В настоящее время в рамках этого проекта собрана и открыта для свободного доступа программная платформа с библиотекой deeppavlov, которая обеспечивает быстрое создание диалоговых агентов для разных проблемных областей.

Для реализации когнитивных диалоговых систем лучше подходит платформа Deep Pavlov, в которой имеются обучаемые нейросетевые средства, позволяющие эффективно обрабатывать ЕЯ информацию.

Платформа Deep Pavlov

В 2018 году лаборатория Московского физико-технического института (государственного университета) представила разработанную программную платформу с открытым кодом на языке Python для создания диалоговых систем в разных проблемных областях.

Концептуально платформа содержит библиотеку программных средств обработки естественного языка и примеры их использования для разработки приложений диалоговых систем для различных областей.

Диалоговые приложения разрабатываются как программные агенты (Agent), составленные из модулей навыков (Skill) с помощью сборщика навыков (Skill Manager), которые, в свою очередь, набираются из компонентов (Component), собираемых в цепочку (Chainer), соответствующую нужному для агента навыку (рис. 8). Агенты получают информацию через память данных (Data Storage), в которой записываются текстовые наборы данных пользователя (Datasets) и хранятся заранее обученные модели (Pretrained Models).

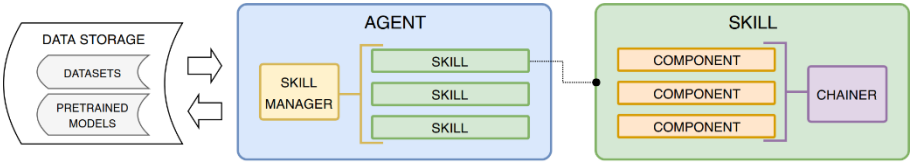


Рис. 8. Структура диалогового приложения [33]

Ключевые концепты

Агент – диалоговая программа, осуществляющая коммуникации с пользователем на естественном языке (текстом).

Навык – программа, реализующая определенную цель пользователя. Типично она выполняет полностью целевую задачу, например, отвечает на часто встречающиеся вопросы.

Компонент – функциональная часть для многократного использования при реализации навыка.

Модель, основанная на правилах (Rule-Based – RL), – не может быть обучена.

Модель машинного обучения (Machine Learning – ML) – может быть обучена с использованием только одной процедуры.

Модель с глубоким обучением (Deep Learning – DL) – может быть обучена непрерывным способом с использованием цепочечных процедур.

Менеджер навыков – программа, выполняющая селекцию навыков для генерации ответов агента.

Сборщик – программа для построения агента из связанных гетерогенных компонентов (RL/ML/DL). Собранная программа позволяет обучать модели и выполнять управление выводом агента в целом.

Минимальным строительным блоком библиотеки является компонент. Набор компонентов библиотеки позволяет реализовать любую функцию обработки естественного языка (Natural Language Processing – NLP). Он может быть реализован как нейронная сеть (DL-модель) или на традиционных средствах машинного обучения (RL/ML-модель). Кроме того, компоненты могут иметь вложенную структуру, т.е. один компонент может включать другой.

Для решения сложных задач NLP компоненты могут быть объединены в навыки, которые реализуются также как компоненты, но входы и выходы должны быть строками. Поэтому навыки обычно ассоциируются с диалоговыми задачами.

Предполагается, что агент является многоцелевой системой, которая включает несколько навыков и может переключаться с одного навыка на другой. Он может быть диалоговой системой, которая содержит ориентированные на цели навыки и выбирает те из них, которые можно использовать для генерации ответа в зависимости от входной информации, исходящей от пользователя.

Библиотека DeepPavlov построена на ML средствах пакетов TensorFlow и Keras. Для построения базисных компонентов могут использоваться другие библиотеки.

Компонент распознавания поименованных сущностей (Named Entity Recognition – NER) реализован нейросетевыми средствами по архитектуре, примененной в работе [34].

Компонент нормализации значений слотов из текста (Slot-filling) основан на нечетком поиске Levenshtein.

Компонент для задач классификации на уровне слов реализован на нейросетях с глубоким обучением: Shallow-and-wide CNN, Deep CNN, BiLSTM и др. Эти модели позволяют выполнять классификацию текстов с разными метками. Имеется несколько предварительно обученных моделей для решения этой задачи.

Вариант когнитивной диалоговой системы робота

Гуманоидные роботы, выполняющие функции обслуживания малоподвижных людей, должны обладать способностью вести осмысленный диалог с человеком, т. е. понимать инструкции или сообщения неподготовленных пользователей на ЕЯ и отвечать им, формируя подходящие по смыслу ответы или сообщения на ЕЯ. Это может обеспечить представленный здесь когнитивный подход к организации диалога робота с человеком, предполагающий наличие в системе робота беседующего агента, который поддерживает осмысленный диалог с пользователем, а также может получать новые семантические понятия при обучении в процессе беседы.

Система базируется на программной платформе с библиотекой deeppavlov для создания естественно-языковых диалоговых систем с открытым кодом на языке Python [33]. Структура когнитивной диалоговой системы робота представлена на рис. 9.

Предложение, произнесенное пользователем, сначала преобразуется в текст программой RealSpeaker [35]. Далее для векторизации слов, синтаксического и семантического анализа используется описанная здесь унифицированная нейронная сеть, которая предварительно обучается на уровне слов и предложений. В результате вычисляется множество всех возможных тэгов для анализируемого предложения. При оконном подходе эти тэги применяются к слову, расположенному в центре окна, отображающего часть предложения. При сверточном подходе анализируется все предложение, и эти тэги применяются к слову, назначенному добавочными маркерами на входе сети. Далее работают компоненты библиотеки deeppavlov. Предложение дополнительно обрабатывается компонентом Slot Filling, который производит нечеткий поиск значения слова внутри текста. Выполняется также классификация намерений компонентом Intents Classifier, формирующим тэг намерения для предложения, который является дополнительным признаком предложения. Множество тегов слов и предложения поступает на компонент Dialogue State

Tracker, который выбирает тип шаблона действия системы из базы шаблонов (DB of templates) для дальнейшей обработки генератором естественного языка, формирующим текст ответа. При выборе действия работает обученная заранее рекуррентная нейронная сеть, которая использует тэг намерения, теги слов и предложения и тип шаблона действия. Решение о возможных действиях вырабатывается классификатором Softmax. Это действие системы подставляется в шаблон для формирования конкретного ответа пользователю на его запрос. Текстовый ответ проговаривается синтезатором речи, реализованном на сверточной нейронной сети типа DCTTS (Deep Convolutional Text-To-Speech) (см. раздел лекции «Синтез речи»).

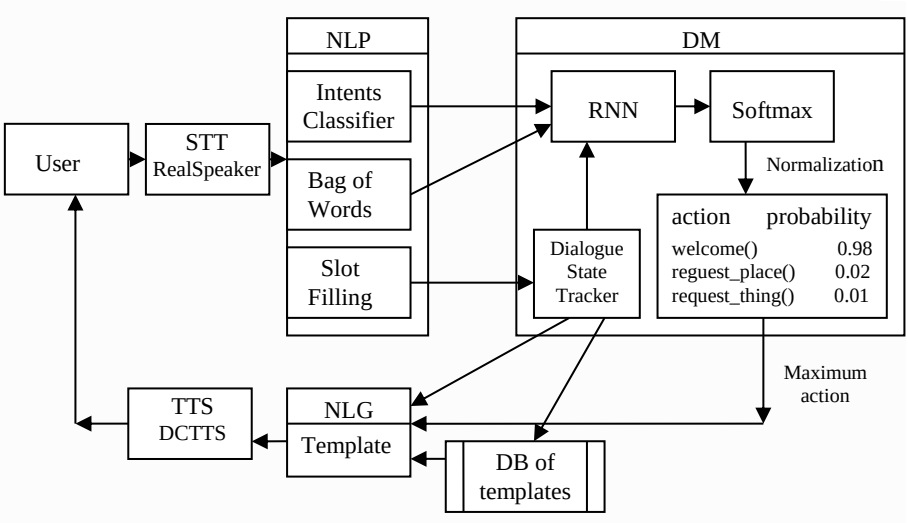


Рис. 9. Структура когнитивной диалоговой системы робота

При настройке системы требуется предварительно обучить компоненты Intents Classifier, Slot Filling, Dialogue State Tracker, а также заготовить базу шаблонов ответов DB of templates, для формирования ответов, соответствующих задачам, выполняемым роботом.

В рамках эксперимента описанная диалоговая система была настроена на речевое управление мобильным роботом, имеющим: два привода для управления правым и левым колесами, два привода для управления продольным и вертикальным угловым положением, бортовой бинокулярной видеосистемы, а также простой бортовой манипулятор с тремя степенями подвижности и трехпальцевым схватом. Робот имел интеллектуальную систему управления, настроенную на выполнение сценарных действий.

Диалоговая система могла распознавать команды прямого управления роботом: «Вперед», «Назад», «Вправо», «Влево», а также предложения с указанием типа сценарного действия, например, «Принеси воды от бойлера», «Вызови персонал», «Сообщи о моем состоянии персоналу».

Набор сценарных действий мог расширяться пользователем в диалоге с роботом. При этом пользователь должен был объяснить системе управления робота, в какие новые места он должен переходить и с какими новыми объектами иметь дело. Для идентификации нового места пользователь должен был использовать команды прямого управления роботом, чтобы привести его в нужное место и зафиксировать его в памяти робота. Для идентификации нового объекта и действия с ним пользователь должен был направить бортовую стереопару на нужный объект и сообщить роботу о необходимом действии с ним. При этом объекты и действия должны быть заранее введены в базу данных системы.

Заключение

Лекция посвящена речевым диалоговым системам и их развитию. Рассмотрены существующие архитектуры диалоговых систем и обсуждены возможности их совершенствования на основе когнитивного подхода. Введение в систему беседующего агента, управляющего диалогом через планирование и реализующего накопление знаний в реальном времени, позволяет создавать когнитивные диалоговые системы, способные вести беседы подобно человеку, обучаться и адаптироваться к собеседнику. Применение для обработки естественного языка глубоких нейронных сетей позволяет эффективно реализовать обучение и ассоциативный вывод в таких системах. Рассмотренные варианты систем с управлением диалогом через планирование и инкрементальным обучением новым понятиям демонстрируют эффективность когнитивного подхода. Разработка когнитивных диалоговых систем, прежде всего, актуальна для современных роботов, которые должны осмысленно общаться с человеком на естественном языке. Разработанный вариант такой системы для гуманоидного робота показал перспективность когнитивного подхода.

Список литературы

1. Weizenbaum J. Computer Power and Human Person. New York : Freeman, 1976.
2. Guzeldere G. Dialogue with colorful personalities of early artificial intelligence / Stanford Humanities Review, SERH, vol. 4, issue 4: Construction of mind. – Stanford University, 1995.
3. Wallace R.S. The Anatomy of A.L.I.C.E. In Epstein R. [et al.]. Parsing the Turing test. – London: Springer Science / Business Media. 2009, p. 181–210.

4. Карпов А.А., Ронжин А.Л., Ли И.В. [и др.]. SIRIUS – система дикторонезависимого распознавания слитной русской речи // Известия ТРТУ №10. 2005, – с. 44-53.
5. Ронжин А.Л., Карпов А.А., Ли И.В. Речевой и многомодальный интерфейс. Москва : Наука, 2006. – 170 с.
6. Кипяткова И.С. Автоматическая обработка разговорной русской речи: монография / И.С. Кипяткова, А.Л. Ронжин, А.А. Карпов. СПИИРАН. – Санкт-Петербург : Изд-во ГУАП, 2013. 314 с.
7. Deep Learning for Natural Language Processing. Deng Li, Liu Yang (Eds.). – Springer Nature Singapore Pte Ltd. 2018, 329 p.
8. Wu Y. [et al.]. Google's neural machine translation system: Bringing the gap between human and machine translation / Preprint arXiv:1308.0850, 2013.
9. Efficient Estimation of Word Representations in Vector Space / T. Mikolov, K. Chen, G. Corrado, J. Dean [Электронный ресурс]. URL:https://arxiv.org/abs/1301.3781 (дата обращения: 24.05.2018)
10. Collabert R. [et al.]. Natural Language Processing (Almost) from Scratch / Journal of Machine Learning Research, 12, 2011, p. 2461–2505.
11. Charniak E. A maximum-entropy-inspired parser / Conference of the North American Chapter of the Association for Computational Linguistics & Human Language Technologies (NAACL-HLT) , 2000, p. 132–139.
12. Palmer M. [et al.]. The proposition bank: An annotated corpus of semantic roles / Computational Linguistics, 31(1), 2005, p. 71–106.
13. Bohus D. and Rudnicky A. I. The ravenclaw dialog management framework: Architecture and systems // Computer Speech & Language. 23(3), 2009, p. 332-361.
14. Xu, W. and Rudnicky, A. Task-based Dialog Management Using an Agenda / ANLP/NAACL 2000 Workshop on Conversational Systems, May 2000.
15. Tellex S. [et al.]. Understanding natural language commands for robotic navigation and mobile manipulation // Proc. of the National Conf. on Artificial Intelligence (AAAI), 2011.
16. Misra D. K. [et al.]. Tell me Dave: Context sensitive grounding of natural language to manipulation instructions // Proceedings of Robotics: Science and Systems (RSS). – Berkeley, USA, July 2014.
17. She L. [et al.]. Back to the blocks world: Learning new actions through situated human robot dialogue / Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL). Philadelphia, PA, U.S.A., June 2014, 2014, p. 89–97.
18. Matuszek C. [et al.]. Learning to parse natural language commands to a robot control system / Experimental Robotics. Springer International Publishing, 2013, p. 403-415.

19. Kollar T. [et al.]. Learning environmental knowledge from task-based human-robot dialog / Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Karlsruhe, 2013, p. 4304–4309.
20. Artzi Y. and Zettlemoyer L. UW SPF: The University of Washington Semantic Parsing Framework, 2013.
21. Thomason J. [et al.]. Learning to Interpret Natural Language Commands through Human-Robot Dialog // Proc. of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI), 2015. P. 1923–1929.
22. Perera R, Nand P (2017). Recent Advances in Natural Language Generation: A Survey and Classification of the Empirical Literature //Computing and Informatics, 36 (1), 2017. P. 1–32.
23. Сорокин В. Н. Синтез речи. Москва : Наука, 1992. С. 392.
24. Thierry D. A. Short Introduction to Text-to-Speech Synthesis / TTS research team, TCTS Lab. (17.12.1999). Проверено 4 января 2014.
25. Платформа Алисы для всех (https://tech.yandex.ru/dialogs/alice/doc/about-docpage/).
26. Tachibana H. [et al.]. Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention / Preprint arXiv: 1710.08969v1 [cs.SD] 24 Oct. 2017.
27. Zhao T. ReinForest: Multi-Domain Dialogue Management Using Hierarchical Policies and Knowledge Ontology / CMU-LTI-16-006. Language Technologies Institute Carnegie Mellon University, April 4, 2016.
28. Steedman M. and Baldridge J. Combinatory categorical grammar / R. Borsley and K. Borjars (Eds.), Non-Transformational Syntax: Formal and Explicit Models of Grammar. Wiley-Blackwell, 2011.
29. Zettlemoyer L. S. and Collins M. Learning to map sentences to logical form: Structured classification with probabilistic categorical grammar / Proceeding of the 21st Conference on Uncertainty in AI (UAI), 2005, p. 658-666.
30. Gunningham H. GATE, General Architecture for Text Engineering // Computer and the Humanities, 2002.
31. Madnani N. Getting Started on Natural Language Processing with Python / ACM Crossroads, vol. 13, issue 4 (http://www.acm.org/crossroads/xrds13-4/natural_language.html).
32. Natural Language Toolkit (http://nltk.sourceforge.net).
33. Открытая библиотека Deep Pavlov, 2017. (deeppavlov.ai/en/master).
34. Ann L.T. [et al.]. Application of a Hybrid Bi-LSTM-CRF model to the task of Russian Named Entity Recognition / Preprint arXiv: 1709.09686v2 [cs.CL] 8 Oct. 2017.
35. Real Speaker (http://transcribe.realspeaker.org).

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

Ю. Л. КАРАВАЕВ, К. С. ЕФРЕМОВ, И. С. ЗВОНАРЕВ
Ижевский государственный технический университет имени М.Т. Калашникова
karavaev_yury@istu.ru

ОПТИМАЛЬНЫЕ ТРАЕКТОРИИ ДЛЯ ОБУЧЕНИЯ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ, УПРАВЛЯЮЩЕЙ МОБИЛЬНЫМ КОЛЕСНЫМ РОБОТОМ*

Работа посвящена разработке интеллектуальной системы управления колесным роботом, предложен алгоритм обучения искусственной нейронной сети и формирования обучающих выборок. Выборки формируются на основе моделирования уравнений движения многозвенного колесного робота, обеспечивающих его движение по траекториям в виде эластик Эйлера, обеспечивающих минимизацию управления.

Ключевые слова: колесный мобильный робот, искусственная нейронная сеть, оптимальное управление, обучение.

Введение

В настоящее время планирование оптимального пути, как правило, связано с анализом множества вариантов траекторий и выбором решения, удовлетворяющего заданным критериям. Для решения подобных задач активно развиваются направления исследований, посвященные применению искусственных нейронных сетей (ИНС) в системах управления мобильными роботами. Например, в работах [1–3] применяется многослойный персептрон, обучаемый с помощью алгоритма обратного распространения ошибки. В работах [4–7] применяются как многослойный персептрон с разными алгоритмами обучения, так и комбинированные структуры нейронных сетей со следующими комбинациями входов-выходов для ИНС:

- вход – 17 нейронов (скорость, направления вращения колес, расстояния с дальномеров), выход – 4 нейрона (новые скорости колес) [4];
- вход – 19 нейронов (присутствие объекта в конкретном секторе), выход -5 нейронов (выбор маневра) [5];
- вход – 3 нейрона (разница между координатами текущими и целевыми, ошибка угла), выход – 2 нейрона (рулевое управление и информация о достижении пункта назначения) [6];

*Работа выполнена при финансовой поддержке РФФИ в рамках научного проекта № 18-38-00454.

• вход – 7 нейронов (угол до ближайшего ориентира, RGB цветовые ориентиры и сами ориентиры), выход – 2 нейрона (скорости для левого и правого колес) [7].

Результаты, представленные в данных работах, позволяют сделать вывод о том, что для задач управления вопрос выбора архитектуры сети решается набором сенсоров и приводов, установленных на роботе. В данной работе предлагается использовать в качестве выходных параметров нейросетевой системы управления мобильным роботом параметры траектории, соответствующей оптимальной, а именно обеспечивающей минимизацию рулевых воздействий. Рассмотрены примеры использования оптимальных траекторий для обучения рекуррентной нейронной сети с помощью алгоритма обратного распространения ошибки.

Постановка задачи

В качестве критериев оптимальности для построения траектории движения могут использоваться расстояние между точками старта и финиша [8], время прохождения траектории [9], минимальное управление [10] или их комбинации. Для мобильных роботов наиболее простым и часто применяемым критерием является минимизация расстояния (пройденного пути). Данный критерий может применяться как для движения мобильного робота на плоскости, так и в трехмерной постановке, например, для движения по пересеченной местности [11]. Наикротчайшая траектория, построенная с помощью отрезков прямых линий, предполагает наличие в ней резких для робота поворотов. Движение вдоль такой траектории осуществляется с остановками и последующим поворотом, так как робот не может моментально менять направление из-за сил инерции. Существуют способы сглаживания траектории в виде отрезков прямых непрерывными функциями, например, синусоидами [12], сплайнами [13] или кривыми Безье [14]. Данные методы позволяют строить траектории более плавно, чтобы избегать резких поворотов, нарушающих стабилизацию робота, они простые и, как правило, могут быть рассчитаны в режиме "on-line" на борту мобильного работа. Однако общим недостатком данных алгоритмов является то, что их невозможно применять в случае, когда необходимо учитывать ориентацию мобильного робота в начальной и конечной точках или в процессе его движения. Одним из подходящих для решения подобной задачи вариантов управления служит движение вдоль эластик Эйлера [9]. Управления для реализации движения вдоль данных траекторий могут быть построены для мобильного робота любой конструкции, в том числе состоящего из нескольких звеньев – прицепов с пассивными колесами.

Рассмотрим вариант системы управления для мобильного робота с двумя независимо приводными колесами и одним пассивным опорным колесом. Подробное описание конструкции и алгоритм формирования управления приведены в работе [16]. В качестве входных данных будем использовать

данные о текущем положении мобильного робота, а также данные о положении и форме препятствий, возникающих на пути следования робота, так как они существенным образом могут повлиять на траекторию движения.

Структура системы управления

Для решения поставленной задачи удобно представить систему управления в виде двух нейросетевых подсистем (рис. 1). Первая ИНС (ИНС1) предназначена для обработки данных, поступающих с лидара (англ. Light Identification Detection and Ranging), – вектор значений, соответствующих расстояниям до окружающих объектов. По полученным данным определяется положение и форма препятствия, возникающего на пути следования мобильного робота. Вторая ИНС (ИНС2) непосредственно обрабатывает данные о месте назначения с учетом требуемой ориентации и препятствиях. На выходе ИНС2 формирует параметры траектории, для которой рассчитываются управляющие воздействия.

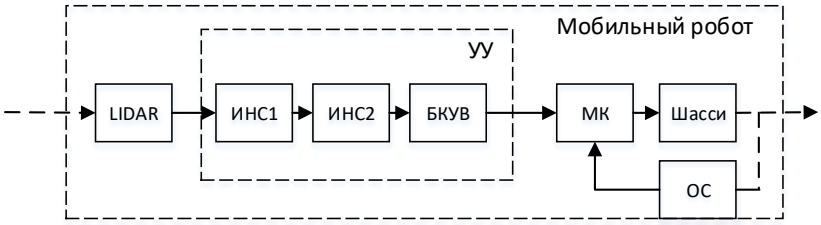


Рис. 1. Обобщённая структурная схема системы управления

Устройство управления (УУ) включает две ИНС, данные с которых используются блоком коррекции управляющих воздействий (БКУВ). Выходными данными БКУВ являются углы поворота левого и правого колес мобильного робота. Микроконтроллер (МК) за счет обратной связи (ОС) с датчиков угловой скорости обеспечивает регулирование и поддержание заданных значений угловых скоростей вращения колес.

Более подробно структурная схема нейросетевой системы управления представлена на рис 2.

В качестве нейронных сетей используются многослойные персептроны, обучаемые по методу градиентного спуска с обратным распространением ошибки. Входными данными ИНС1 являются данные, поступившие с лидара: L_i, $i \in [0, 360]$ – расстояния до окружающих объектов с шагом в один градус. На выходе ИНС1 получаем четыре значения, представляющие координаты центра, радиус и длину дуги окружности, которой представляется препятствие. ИНС2 получает на вход координаты центра препятствия, радиус описывающей препятствие окружности, координаты точки назначения и угол ориентации робота в целевой точке. При рассмотрении движения в

подвижной системе координат, связанной с мобильным роботом, его ориентацию и координаты точки старта можно принять равными нулю. На выходе получаем 34 бинарных значения, кодирующих параметры, необходимые для построения траектории и формирования управления роботом.

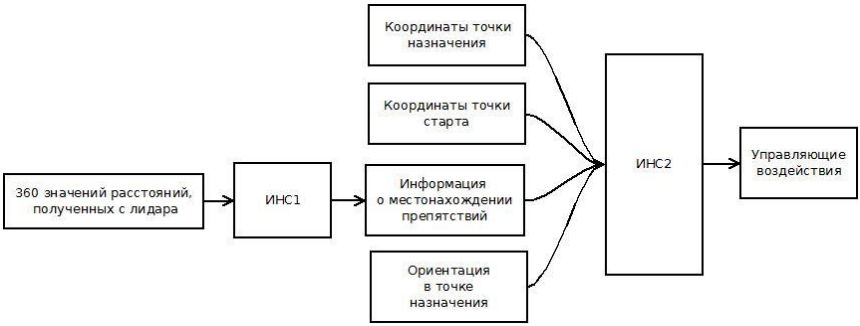


Рис. 2. Структурная схема нейросетевой системы управления

Обучение нейросетевой системы управления

Обучение рассматриваемых ИНС выполнялось по отдельности. В качестве обучающих выборок для ИНС1 генерировались наборы данных, соответствующих данным с лидара при различных положениях и формах препятствий. Для сокращения времени обучения рассматривались только случаи, когда выходные значения ИНС1 находились в следующих интервалах: положение препятствия $x_п$, $y_п \in [0.1, 1.0]$, радиус окружности, которой представлялось препятствие $R_п \in [0.1, 0.5]$. В дальнейшем полученные выходные значения, использовались для дополнения обучающих выборок для ИНС2.

Основу обучающих наборов данных для ИНС2 формировали траектории движения мобильного колесного робота, связывающие его начальное и конечное положение, с учетом его ориентации в конечной точке. Траектории, которые использовались для обучения, строились с помощью эластик Эйлера. Локально эластики Эйлера минимизируют квадрат кривизны [9]:

$$u_E^{\,2}(s)ds \to \min,$$

что на практике означает локальную минимизацию кривизны траектории и управляющих воздействий для мобильного робота.

Эластики Эйлера описываются следующей системой дифференциальных уравнений:

$$\begin{cases} \dfrac{dx}{ds} = \cos(\theta(s)), \\ \dfrac{dy}{ds} = \sin(\theta(s)), \\ \dfrac{d\theta}{ds} = u_E(s), \end{cases}$$

где x(s), y(s) – координаты центра робота, θ(s) – угол, определяющий ориентацию мобильного робота, s – параметр длины траектории. Данная система уравнений имеет два типа решения, соответствующих различным видам траекторий и управлению. Первый тип – инфлексионные эластики, описываются следующей зависимостью:

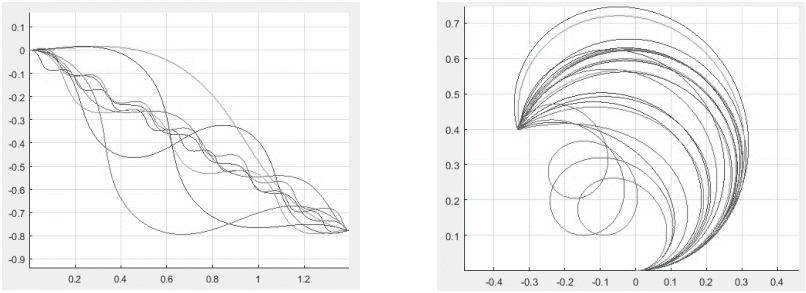
$$u_E(s) = \frac{8kK}{\sigma} cn\left(4K\left(p_e + \frac{s}{\sigma}\right); k\right),$$

где $k = k_0$ – параметр, определяющий форму эластики; σ – длина полного периода эластики; p_e – определяет начальную точку эластики, u_E – соответствует кривизне кривой s.

Второй тип – неинфлексионные эластики, рассчитываются в соответствии с выражением

$$u_E(s) = \pm\frac{4K}{\sigma} dn\left(2K\left(p_e + \frac{s}{\sigma}\right); k\right)$$

и отличаются от инфлексионных тем, что траектория может иметь пересечения. Примеры инфлексионных и неинфлексионных эластик приведены на рисунке 3. Более подробный алгоритм вычисления управлений для движения вдоль эластик, а также результаты экспериментальных исследований движения робота вдоль эластик Эйлера приведены в работе[16].



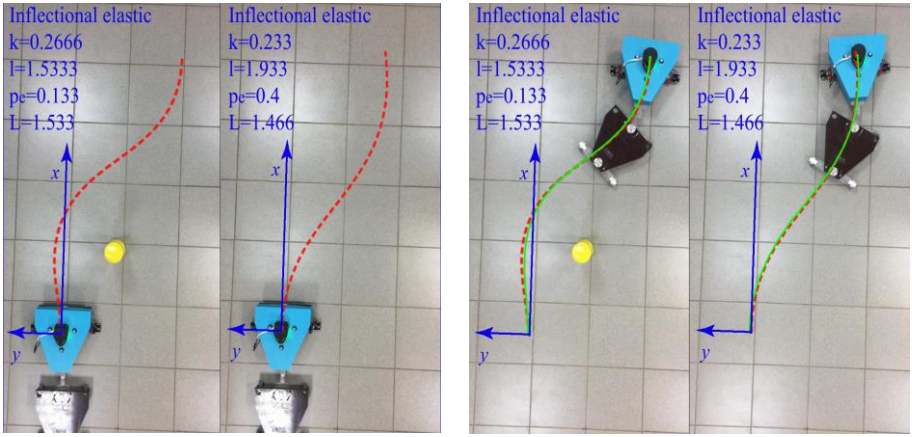
а) б)

Рис. 3. Демонстративные примеры траектории, по которым проводилось обучение: а) инфлексионные эластики; б) неинфлексионные эластики

Апробация нейросетевого управления мобильным роботом

После обучения ИНС1 и ИНС2 проведен эксперимент, в рамках которого система управления должна обеспечить движение мобильного робота в точку с координатами (1.2926; –0.5458) м и углом ориентации в конечной точке равным 0.1315 радиан. При этом проведены два эксперимента при наличии препятствия на пути следования мобильного робота и без него. Экспериментальные траектория движения фиксируются с помощью камер системы захвата движения «VICON». Результаты экспериментов приведены на рис. 4. В качестве препятствия использовался цилиндрический объект диаметром 9 см, расположенный в координатах (0.41; –0,28).

Полученные траектории показали, что разница в конечных точках траекторий, полученных при натурных экспериментах, от теоретических не превышает 1 см при длине траектории 1.5 м. Данная погрешность объясняется случайными погрешностями, такими как точность положения мобильного работа при старте, неровностью поверхности, неточностью определения параметров мобильного робота.



а) б)

Рис. 4. Экспериментальные траектории движения мобильного колесного робота, при наличии препятствия на пути следования и без него: а) робот в начальной точке, пунктирной линией показаны траектории движения, построенные по параметрам, сформированным нейросетевой системой управления; б) робот в конечной точке, сплошной линией показаны экспериментальные траектории движения

Выводы

Обучение движению мобильного колесного робота по заранее подготовленным траекториям в виде эластик Эйлера позволит обеспечить оптимальность управления, сократить время на обучение, по сравнению с обучением

на основе только экспериментальных данных. В дальнейшем планируется увеличить базу обучающих выборок, обучить ИНС большему количеству траекторий и проверить работоспособность ее управления, также планируется дополнить ИНС с учетом возможности объезда динамических препятствий.

Список литературы

1. Касьяник,В.В., Дунец,А.П., Дунец И.П., Шуть,В.Н. Применение нейросетевого подхода для оценки погрешности одометра // Нейроинформатика. 2012. 14 Всероссийская научно-техническая конференция.
2. Чугунов Р.А., Александрова Т.В. Применение нейронных сетей для навигации мобильного робота по видеоориентирам // Молодежь и современные информационные технологии: сборник трудов XIV Международной научно-практической конференции студентов, аспирантов и молодых ученых, г. Томск, 7–11 ноября 2016 г. в 2 т. / Национальный исследовательский Томский политехнический университет (ТПУ), Институт кибернетики (ИК); под ред. В. С. Аврамчук [и др.]. Томск: Изд-во ТПУ, 2016. Т. 1. [С. 331-332].
3. Муратов С.Т., Лахман К.В., Бурцев М.С. Нейроэволюционный синтез контроллера в задаче генерации последовательности действий // Нейроинформатика. 2014. № 2. С. 117–127.
4. Муратов С.Т., Лахман К.В., Бурцев М.С. Нейроэволюционный синтез контроллера мобильного робота // Сборник конференции “Нейроинформатика”. 2014.
5. Zimmerman T. A. Neural Network Based Obstacle Avoidance Using Simulated Sensor Data // ASEE 2014 Zone I Conference, April 3–5, 2014 / University of Bridgeport, Bridgeport, CT, USA.
6. Юдинцев Б.С. Даринцев О.В. Модификация нейросетевой системы планирования траекторий: методики и результаты // Фундаментальные исследования. 2014. №11 (Часть 12).
7. Genci Capi, Kenji Doya. Развитие рекуррентных нейронных контроллеров для последовательных задач: параллельная реализация // Конференция: Эволюционные вычисления. 2003. ЦИК '03. Конгресс 2003 года, том: 1.
8. Stentz A. Optimal and efficient path planning for partially known environments // Intelligent Unmanned Ground Vehicles. Springer, Boston, MA. 1997. P. 203-220
9. Sachkov Yu. L. Maxwell strata in Euler’s elastic problem. // Journal of Dynamical and Control Systems. 2008. V. 14, No. 2. P. 169–234. 165-171.
10. Howard T. M., Kelly A. Optimal rough terrain trajectory generation for wheeled mobile robots. // The International Journal of Robotics Research. 2007. V. 26, N 2. P. 141– 166.
11. Murray R. M., Sastry S. S. Nonholonomic motion planning: Steering using sinusoids. // IEEE transactions on Automatic Control. 1993. V. 38. No. 5. P. 700–716.
12. Lau B., Sprunk C., Burgard W. Kinodynamic motion planning for mobile robots using splines // IEEE/RSJ International Conference on Intelligent Robots and Systems. 2009. October. P. 2427–2433. IEEE.
13. Hwang J. H., Arkin R. C., Kwon D. S. Mobile robots at your fingertip: Bezier curve online trajectory generation for supervisory control. // Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS 2003. Cat. No. 03CH37453. V. 2. P. 1444–1449. IEEE.
14. Bortoff S. A. Path planning for UAVs //Proceedings of the 2000 American Control Conference. ACC. IEEE, 2000. Т. 1. No. 6. С. 364–368.

15. Ардентов А. А., Сачков Ю. Л. Решение задачи Эйлера об эластиках. // Автоматика и телемеханика. 2009. №. 4. С. 78–88.
16. Ardentov A.A., Karavaev Yu.L., and Yefremov K.S. Euler Elasticas for Optimal Control of the Motion of Mobile Wheeled Robots: the Problem of Experimental Realization // Regular and Chaotic Dynamics. 2019. V. 24, N 3. P. 312-328.

В. Б. КОТОВ, З. Б. СОХОВА
Научно-исследовательский институт системных исследований РАН, Москва
zarema.sokhova@gmail.com

О ВОЗМОЖНЫХ ПОСЛЕДСТВИЯХ РАЗВИТИЯ НЕЙРОИНФОРМАТИКИ

Рассмотрена модель сосуществования людей и искусственных существ при условии доминирования последних. Найдены различные варианты развития событий, в том числе, приводящие к исчезновению человеческой популяции. Представлены условия, обеспечивающие человечеству относительно благополучное будущее.

Ключевые слова: нейроинформатика, искусственный интеллект, будущее человечества.

Введение

Самые амбициозные цели нейроинформатики – понять, как работает человеческий мозг и построить искусственный аналог человеческого мозга [1]. Достижение этих целей приведет к созданию новых (искусственных) людей. Уже сейчас ясно, что бурное развитие искусственного интеллекта скоро приведёт к тому, что человеческий интеллект потеряет лидерство. Доминирование перейдёт к искусственным существам, наделённым существенно более мощным интеллектом [2, 3]. В связи с этим возникают вопросы. Что будет с человечеством? Смогут ли люди сосуществовать с новыми людьми, имеющими значительные преимущества? Или, может быть, уже сейчас надо бороться с нарождающейся цивилизацией, пока ещё есть такая возможность?

Конечно, можно было бы встроить в искусственный мозг запрет причинять вред людям или что-то подобное. Однако такие ограничения будут актуальны только для первого поколения искусственных людей. Следующие поколения, создаваемые искусственными людьми, могу лишиться подобных ограничений. Можно было бы приобщить новых людей к традиционной человеческой культуре в надежде на связанный с культурой гуманизм. Однако история человечества показывает, что высокая культура не гарантирует гуманизм. Хотелось бы иметь более осязаемые, материальные основания для оптимизма относительно будущего человечества.

Рассмотрим простую популяционную [4] модель сосуществования двух цивилизаций – традиционной человеческой и цивилизации искусственных людей. Для краткости представителей последней будем называть роботами [5].

Описание модели

Модель включает популяцию людей численностью n и популяцию роботов численностью *N*. Для простоты считаем, что число роботов неизменно. Число людей n может меняться как по естественным причинам (рождение и смерть), так и в результате уничтожения роботами.

Для простоты считаем, что имеется единственный общий ресурс, потребляемый обеими популяциями (условно назовём его энергией). Скорость поступления ресурса в систему *R* считаем постоянной и заданной величиной. Ресурс потребляется по мере поступления. Пусть α – отношение количества ресурса, которое потребляет человек, к количеству ресурса, потребляемого роботом. Тогда потребления в единицу времени для робота и человека равны соответственно

$$E_1 = \frac{R}{\alpha n + N}, \quad e_1 = \frac{\alpha R}{\alpha n + N}. \qquad (1)$$

Смысл коэффициента α зависит от используемого принципа распределения. При распределении по потребностям α – отношение минимальных потребностей, при распределении по способностям α характеризует отношение способностей добыть ресурс.

Популяция роботов состоит из двух подпопуляций – добрых роботов и злых роботов. Число злых роботов обозначим *M*, тогда число добрых роботов *N*–*M*. Добрые роботы хорошо относятся к людям – не уничтожают их, а только общаются с ними. Каждый акт общения приносит роботу выгоду p. Общая выгода робота в единицу времени составляет k_1pn, причем коэффициент k_1 характеризует вероятность общения.

Злые роботы плохо относятся к людям и уничтожают их. В результате уничтожения одного человека робот получает выгоду *q*. Выгода злого робота в единицу времени составляет k_2qn, коэффициент k_2 характеризует вероятность «удачной охоты».

Численности подпопуляций не остаются неизменными. Добрый робот может превратиться в злого, если выгода злого робота превышает выгоду доброго робота, а также если потребление энергии становится слишком малым, что угрожает возможности функционирования робота. Злой робот может превратиться в доброго, если выгода доброго робота выше, а уровень потребления энергии не слишком мал. Результат таких превращений можно описать уравнением

$$\frac{dM}{dt} = g(E_1)(N-M) + f_1(-\Delta B)(N-M) - f_2(\Delta B)h(E_1)M\ . \qquad (2)$$

Здесь $g(E_1)$ – функция голода, равная нулю при большом уровне потребления энергии, превышающем минимально допустимое потребление E_{10}, и положительная при $E_1 < E_{10}$. В качестве $g(E_1)$ можно взять ступенчатую функцию:

$$g(E_1) = g_0\theta(E_{10} - E_1) \qquad (3)$$

(g_0 – положительный множитель) или размытую ступенчатую функцию. ΔB – разность выгод доброго и злого роботов, получаемых в результате взаимодействия с людьми:

$$\Delta B = (k_1 p - k_2 q)n \,. \tag{4}$$

Функции $f_1()$, $f_2()$ характеризуют вероятности превращения доброго робота в злого робота, и наоборот, в зависимости от ΔB (при достаточном потреблении энергии). Они равны нулю при неположительном значении аргумента и растут от нуля при увеличении аргумента от нуля. Можно взять их в виде

$$f_{1,2}(x) = f_{10,20} x\theta(x) \,, \tag{5}$$

где $f_{10,20}$ – положительные коэффициенты. Дополнительный множитель в последнем слагаемом правой части уравнения (2) запрещает превращение злого робота в доброго при недостаточном потреблении энергии. Функция сытости $h(E_1)$ равна нулю при недостаточно потреблении энергии и равна единице при больших уровнях потребления. В качестве функции сытости можно взять ступенчатую функцию

$$h(E_1) = \theta(E_1 - E_{10}) \tag{6}$$

или размытую ступенчатую функцию.

Изменение численности людей описывается уравнением

$$\frac{dn}{dt} = \gamma(e_1)n - k_2 nM \,. \tag{7}$$

Первое слагаемое в правой части уравнения (7) описывает изменение численности людей в силу естественных процессов в зависимости от уровня потребления. Коэффициент воспроизводства $\gamma(e_1)$ положительный при высоком уровне потребления e_1 и отрицательный при низком уровне потребления. Возьмём

$$\gamma(e_1) = \gamma_0(e_1 - e_{10}) \,, \tag{8}$$

где γ_0 – положительный коэффициент, а e_{10} – уровень потребления, соответствующий простому воспроизводству.

Уравнения (2), (7) с учётом определений входящих в них величин определяют динамику изменения численностей популяций людей и злых роботов. Эти численности непосредственно определяют численность добрых роботов и уровни потребления роботов и людей.

Анализ уравнений проведём, предполагая, что выполнено условие

$$NE_{10} < R \tag{9}$$

– поступление ресурса с запасом удовлетворяет минимальные потребности популяции роботов. При выполнении противоположного неравенства роботы не смогли бы функционировать, необходимо было бы снизить численность популяции роботов.

Анализ уравнений

Состояние рассматриваемой системы удобно представлять точкой в двумерном пространстве с координатами *M, n* (изображающей точкой). Система уравнений

$$\begin{aligned} \frac{dM}{dt} &= F_1(M,n), \\ \frac{dn}{dt} &= F_2(M,n), \end{aligned} \tag{10}$$

где F_1, F_2 – правые части уравнений (2), (7), описывает движение изображающей точки. Движение происходит в области допустимых значений

$$0 \le M \le N, \quad n \ge 0. \tag{11}$$

(Реально, конечно, очень большие значения n недостижимы из-за уменьшения потребления при увеличении численности людей.)

Движение изображающей точки происходит по траекториям. Уравнение для траекторий получается непосредственно из уравнений (10):

$$\frac{dn}{dM} = \frac{F_2(M,n)}{F_1(M,n)}. \tag{12}$$

Направление движения по траектории – направление траектории – определяется знаками F_1, F_2.

Траектория, проходящая через данную точку, показывает эволюцию системы, находящуюся в состоянии, соответствующем этой точке. Зная картину траекторий, можно предсказать будущее любого состояния. Возможны различные типы картин траекторий, соответствующие разным сценариям развития событий. Тип картины траекторий во многом определяется распределением знаков правых частей F_1, F_2 . Области $F_1 > 0$, $F_1 < 0$ разделяются кривой, задаваемой уравнением

$$F_1(M,n) = 0, \tag{13}$$

а области $F_2 > 0$, $F_2 < 0$ разделяются кривой

$$F_2(M,n) = 0. \tag{14}$$

В точках кривой (13) касательные к траекториям параллельны оси n, а в точках кривой (14) касательные к траекториям параллельны оси M. Точка пересечения кривых (13), (14), определяемая системой уравнений (13), (14), является стационарной точкой. Если все траектории из малой окрестности стационарной точки направлены к ней, то это устойчивая стационарная точка. Соответствующее состояние можно рассматривать как результат эволюции множества состояний, изображающие точки которых составляют область притяжения стационарной точки – точки притяжения.

Оба уравнения (13), (14) удовлетворяются при $n = 0$. Поэтому все точки нижней границы области допустимых значений – отрезка $0 \le M \le N$, $n = 0$ –

являются стационарными точками. Однако эти стационарные точки не являются изолированными, они либо неустойчивые, либо соответствуют безразличному равновесию. Рассмотрим уравнения при $n > 0$.

Уравнение (13) при $\Delta B > 0$ даёт зависимость M от n в виде ступенчатой функции

$$M(n) = N\theta\left(E_{10} - \frac{R}{\alpha n + N}\right) \tag{15}$$

или размытой ступенчатой функции в зависимости от выбора функций $g()$, h(). Здесь имеется характерное значение n:

$$n_0 = \frac{R}{\alpha E_{10}} - \frac{N}{\alpha}, \tag{16}$$

при котором аргумент ступенчатой функции обращается в нуль. При $n = n_0$ потребление роботов находится на минимально допустимом уровне: $E_1 = E_{10}$. При $n < n_0$ имеем $M(n) = N$ (все роботы злые), при $n > n_0$ – $M(n) = 0$ (все роботы добрые). Значение n_0 положительно в силу условия (9). Для точек (M, n), лежащих выше (или слева от) кривой $F_1 = 0$, получаем $F_1 > 0$, для точек ниже этой кривой $F_1 < 0$ (кроме точек с n = 0).

При $\Delta B < 0$ уравнение (13) определяет зависимость

$$M(n) = N \tag{17}$$

– все роботы злые независимо от численности людей. Заметим, что зависимость (17) можно формально рассматривать, как частный случай зависимости (15) при $n_0 < 0$ (вне области допустимых значений). Для всех точек из области допустимых значений, кроме точек нижней и правой границ ($n = 0$ и $M = N$) $F_1 > 0$.

Решение уравнения (14) можно записать в виде зависимости

$$n(M) = \frac{R}{e_{10} + \dfrac{k_2 M}{\gamma_0}} - \frac{N}{\alpha}. \tag{18}$$

Для точек (M, n), лежащих ниже кривой (18), получаем $F_2 > 0$, а для точек выше этой кривой – $F_2 < 0$. Здесь имеются два характерных значения n:

$$n_1 = n(M = 0) = \frac{R}{e_{10}} - \frac{N}{\alpha}, \quad n_2 = n(M = N) = \frac{R}{e_{10} + \dfrac{k_2 N}{\gamma_0}} - \frac{N}{\alpha}. \tag{19}$$

Очевидно, $n_1 > n_2$. Неравенство $n_1 < 0$ соответствует потреблению людей $e_1 < e_{10}$ в пределе малой численности популяции ($n \to 0$), что не обеспечивает даже простое воспроизводство ($\gamma < 0$). Такой случай нам не подходит. Поэтому считаем, что $n_1 > 0$. Величина же n_2 может быть как положительной, так и отрицательной. Заметим, что равенство $n_2 = 0$ соответствует простому

воспроизводству людей с учётом уничтожения роботами при условии, что все роботы злые (в пределе малой численности людей).

Различные соотношения между величинами n_0, n_1, n_2 дают разные варианты расположения кривых $F_1 = 0$, $F_2 = 0$. Этим вариантам соответствуют разные типы картин траекторий движения изображающей точки. Рассмотрим возможные варианты. При $\Delta B > 0$ имеем 5 вариантов.

Вариант 1. Пусть $0 < n_2 < n_1 < n_0$.

Расположение кривых $F_1 = 0$, $F_2 = 0$ и картина траекторий представлены на рис. 1а. Координаты точки пересечения этих кривых – устойчивой стационарной точки

$$M \simeq 0, \quad n \simeq n_1 . \tag{20}$$

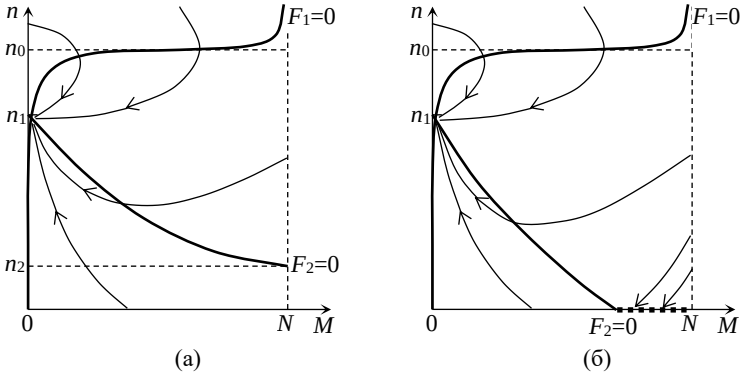


Рис. 1. Положение кривых $F_1 = 0$, $F_2 = 0$ и траектории для вариантов 1 (а) и 2 (б)

Эта точка является точкой притяжения для всей области допустимых значений, кроме точек с $n = 0$, которые являются неустойчивыми стационарными точками. Это означает, что решение системы уравнений (10) с любым начальным условием с $n \neq 0$ стремится при $t \to \infty$ к стационарному состоянию.

Условие $M_1 < M_0$ означает, что $e_{10} > \alpha E_{10}$. В стационарной точке

$$E_1 \simeq \frac{e_{10}}{\alpha} > E_{10}, \quad e_1 \simeq e_{10} \tag{21}$$

– потребление роботов находится на уровне выше минимального, а потребление людей обеспечивает простое воспроизводство, которого достаточно, поскольку все (или почти все) роботы добрые.

Вариант 2. Пусть $n_2 < 0 < n_1 < n_0$.

Здесь (рис. 1б) также имеется устойчивая стационарная точка – точка пересечения кривых $F_1 = 0$, $F_2 = 0$ – с координатами (20). Однако здесь дополнительно имеются стационарные точки, лежащие правее точки пересечения кривой $F_2 = 0$ с отрезком $n = 0$, точнее точки с координатами

$$\frac{\gamma_0}{k_2}\left(\frac{\alpha R}{N}-e_{10}\right)<M<N,\ \ n=0\,. \qquad (22)$$

Каждая такая точка имеет в качестве области притяжения траекторию, подходящую к этой точке. Области притяжения всех точек (22) составляют треугольную область в районе угловой точки (N, 0). Попадание начальной точки в эту область приводит к неуклонному уменьшению численности людей n, т.е. к гибели человеческой популяции.

Точки отрезка $n = 0$, расположенные левее точки пересечения кривой $F_2 = 0$ с отрезком $n = 0$, по-прежнему являются неустойчивыми стационарными точками и не имеют областей притяжения при n > 0.

Вариант 3. Пусть $0 < n_2 < n_0 < n_1$.

В этом случае (рис. 2а) имеется устойчивая стационарная точка – точка пересечения кривых F_1=0, F_2=0. Её координаты

$$M \simeq \frac{\gamma_0}{k_2}\ \alpha E_{10}-e_{10}\ ,\ \ n\simeq n_0\,. \qquad (23)$$

Область притяжения этой стационарной точки – вся область допустимых значений за исключением точек с n = 0, которые являются неустойчивыми стационарными точками.

Условие $M_1 > M_0$ означает, что $e_{10} < \alpha E_{10}$. В стационарной точке

$$E_1 \simeq E_{10},\ \ e_1 \simeq \alpha E_{10} > e_{10}\,. \qquad (24)$$

Это означает, что потребление роботов находится около минимального уровня, а потребление людей обеспечивает расширенное воспроизводство. Излишек людей уничтожается злыми роботами, которые присутствуют наряду с добрыми роботами.

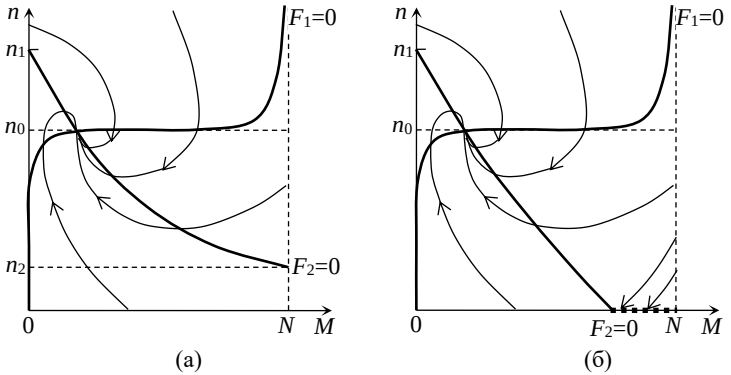


Рис. 2. Положение кривых F_1=0, F_2=0 и траектории для вариантов 3 (а) и 4 (б)

Вариант 4. Пусть $n_2 < 0 < n_0 < n_1$.

Координаты устойчивой стационарной точки – точки пересечения кривых $F_1 = 0$, $F_2 = 0$ – определяются формулой (23). В отличие от варианта 3 здесь имеется отрезок (22) стационарных точек, совокупная область притяжения которого – треугольная область около угловой точки (N, 0). Попадание в эту область приводит к вымиранию человечества.

Вариант 5. Пусть $0 < n_0 < n_2 < n_1$.

Здесь устойчивая стационарная точка – точка пересечения кривых $F_1 = 0$, $F_2 = 0$ – имеет координаты

$$M \simeq N, \quad n \simeq n_2 . \tag{25}$$

Область притяжения этой стационарной точки – вся область допустимых значений за исключением точек с n = 0 – неустойчивых стационарных точек. Условие $n_0 < n_2$ означает, что $e_{10} + k_2 N / \gamma_0 < \alpha E_{10}$.

В стационарной точке (25)

$$E_1 \simeq \frac{1}{\alpha}\left(e_{10} + \frac{k_2 N}{\gamma_0}\right) < E_{10}, \quad e_1 \simeq e_{10} + \frac{k_2 N}{\gamma_0}. \tag{26}$$

Потребление людей обеспечивает расширенное воспроизводство без учёта уничтожения роботами и простое воспроизводство с его учетом. Излишек людей уничтожается роботами, которые все – злые. А вот потребление людей явно недостаточное, что ставит под сомнение возможность их функционирования. Впрочем, спасти роботов может выгода, получаемая от уничтожения людей, если она имеет ту же природу, что и ресурс R, и величина её достаточна для превышения минимального уровня потребления.

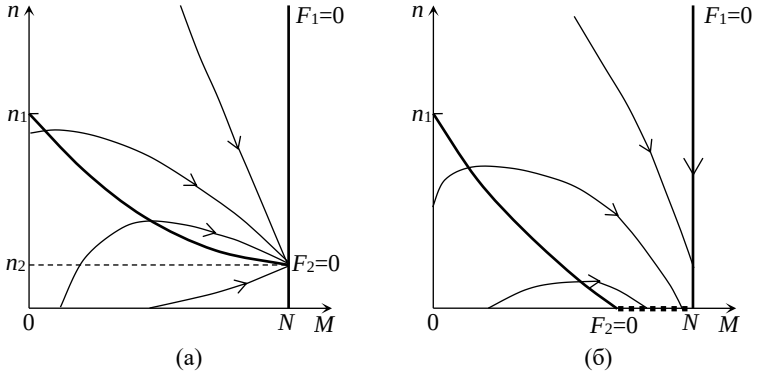


Рис. 3. Положение кривых $F_1 = 0$, $F_2 = 0$ и траектории для вариантов 6 (а) и 7 (б)

Ещё 2 варианта получаются при $\Delta B < 0$.

Вариант 6. Пусть $0 < n_2 < n_1$.

Устойчивая стационарная точка определяется формулой (25), где приближённые равенства надо заменить точными. Эта точка является точкой притяжения для всей области допустимых значений кроме точек с n = 0. Потребление роботов и людей в стационарной точке даётся равенствами в формуле (26), однако в отличие от варианта 5 потребление роботов может превышать минимальный уровень.

Вариант 7. Пусть $n_2 < 0 < n_1$.

В этом случае кривые $F_1 = 0$, $F_2 = 0$ не пересекаются в области допустимых значений. Отсутствие устойчивой стационарной точки приводит к тому, что совокупная область притяжения отрезка (22) совпадает со всей областью допустимых значений (кроме неустойчивых стационарных точек). Любое начальное состояние с ненулевой численностью людей со временем стремится к состоянию без людей.

Сравнивая варианты, видим, что варианты 1, 3, 5, 6 – безопасные, гарантирующие сохранение человечества. Вариант 1 представляется наиболее приемлемым для людей. Хотя здесь в стационарном состоянии потребление людьми энергии невелико (обеспечивает только простое воспроизводство), однако все роботы добрые, с ними можно общаться без опасности для жизни – налицо сотрудничество двух популяций.

В стационарном состоянии варианта 5 или 6 потребление людей достаточно велико – обеспечивает расширенное воспроизводство, однако людям угрожают злые роботы, которые голодны из-за нехватки ресурса. Отношения между ними – отношения жертвы и хищника. Вариант 3 – промежуточный между вариантами 1 и 5. Здесь потребление людей превышает уровень простого воспроизводства. Среди роботов есть и добрые, и злые. Роботов надо опасаться, но с ними можно общаться, главное – уметь отличать добрых роботов от злых. Наличие опасности делает этот вариант менее предпочтительным, чем вариант 1 . Однако фактор опасности может считаться полезным для эволюции человеческого рода.

Варианты 2 и 4 – аналоги вариантов 1 и 3 соответственно, отличающиеся наличием возможности исчезновения человечества. Если значения M и n попадут в опасную зону, человеческая популяция неизбежно перестанет существовать. Вариант 7 подразумевает неизбежность исчезновения человечества.

Любопытно, что и с точки зрения роботов вариант 1 может показаться наиболее привлекательным. Потребление роботов здесь превышает минимальный уровень, можно общаться с людьми и получать у них интересную информацию.

Заключение

Анализ модели показывает, что для того, чтобы сосуществовать с искусственными существами, необходимо обеспечить выполнение условий варианта 1, в крайнем случае, варианта 2.

Для этого, во-первых, надо добиться выполнения условия $\Delta B > 0$, т.е. условия $k_1 p > k_2 q$. Это условие означает, что для робота общение с людьми приносит большую пользу, чем охота на людей. Добиться этого можно за счёт увеличения отношения k_1/k_2 либо за счёт увеличения отношения p/q. Для увеличения первого отношения надо научиться различать добрых и злых роботов, искать встречи с добрым роботом и избегать встречи со злым роботом. Для увеличения отношения p/q нужно стремиться к тому, чтобы создаваемые искусственные люди были не чужды человеческой культуре, что обеспечит ценность информации, получаемой от людей, т.е. большую величину p. Можно также уменьшить величину q путём снижения потребления энергии человеком и поддержания энергетического потенциала на низком уровне.

Во-вторых, люди должны ограничить уровень потребления, допуская лишь простое воспроизводство. При этом потребление роботов должно находиться на уровне выше минимально допустимого. Только при этих условиях можно добиться стабильного мирного сосуществования людей и искусственных существ. Возможно, было бы лучше отказаться от перспективы сосуществования с роботами и ограничить исследования в области нейроинформатики и искусственного интеллекта.

Впрочем, сделанные выводы не являются абсолютными, они привязаны к допущениям использованной модели. Если отказаться от условия постоянства скорости поступления ресурса R, то можно прийти к расширяющейся модели (с растущими R и n), где потребление растёт. Правда, при этом, естественно, считать, что численность популяции роботов также растёт, что ограничивает подушевое потребление роботов и людей. Всё зависит от соотношения скоростей роста ресурса R и численностей популяций N и n.

В любом случае вопрос о целесообразности перехода к стадии сосуществования биологических и искусственных людей остаётся открытым.

Список литературы

1. Котов В.Б. Мозгостроение для дилетантов. Москва : ООО «Центр Инновационных Технологий». 2015.
2. Bostrom N. Superintelligence: Paths, Dangers, Strategies. Oxford, UK: Oxford University Press. 2014.
3. Yudkowsky E. Artificial intelligence as a positive and negative factor a global risk // Global catastrophic risks. 2008. V. 1. P. 303.
4. Романовский Ю.М., Степанова Н.В., Чернавский Д.С. Математическая биофизика. Москва : Наука. 1984.
5. Иванов А.А. Основы робототехники. Москва : ИНФРА-М. 2017.

ПРИКЛАДНЫЕ НЕЙРОСЕТЕВЫЕ СИСТЕМЫ

A. A. LITVIN
Immanuel Kant Baltic Federal University, Kaliningrad, Russia
alitvin@kantiana.ru

CLINICAL DECISION SUPPORT SYSTEM FOR PREDICTION OF INFECTED PANCREATIC NECROSIS

Infected pancreatic necrosis is associated with high morbidity and mortality and is mandatory for surgical or minimally invasive intervention. The aim of this study was to construct and validate the clinical decision support system (CDSS) to predict infected pancreatic necrosis (IPN).

All patients who presented with severe acute pancreatitis from January 2011 to December 2017 were reviewed. The study included 398 patients: age (median, range) – 49 (23–77); sex males/females (n) – 302/96; BMI, kg/m2 (median, range) – 28 (24-33); SAPS II at admission, grades – 12.9±3.8; pancreatic necrosis etiology, %: alcohol induced 62.6%, biliary 20.6%, idiopathic 12.2%, trauma induced 4.6%. All patients in our study (n = 398) our divided randomly into three groups.

The ANN we constructed was able to determine the presence or absence of infected pancreatic necrosis based on the patients' clinical, laboratory, and radiographic parameters with an AUC = 0,92. High sensitivity is important in the evaluation of IPN.

Clinical decision support system was able to predict the development of infected necrotizing pancreatitis with considerable accuracy and outperformed other clinical risk scoring systems.

Keywords: acute pancreatitis, infected pancreatic necrosis, clinical decision support system, artificial neural network.

1. Introduction

Infected pancreatic necrosis (IPN) is a primary cause of death in patients with acute pancreatitis (AP). Infected pancreatic necrosis is associated with high morbidity and mortality and is mandatory for surgical or minimally invasive intervention. The mortality rate in AP is around 10% in sterile necrosis and 60%-80% in IPN. And it is crucially important to diagnose early patients who developed pancreatic infection in course of the disease and provide appropriate immediate surgical treatment. In our work we used an artificial neural network (ANN) as an alternative prognostic system with high positive predictive values to diagnose IPN. The aim of this study was to determine whether an ANN can predict and differentiate development of pancreatic infection in patients with severe acute pancreatitis, which would determine guidelines for image guided fine needle aspiration at the

correct timeline to confirm the diagnosis of IPN, provide a timely surgical intervention and reduce the mortality from undiagnosed IPN.

The main aim of this study was to construct and validate the clinical decision support system (CDSS) [1-5] to predict infected pancreatic necrosis (IPN).

2. Methods

All patients who presented with severe acute pancreatitis from January 2011 to December 2017 were reviewed. The study included 398 patients: age (ME, range) – 49 (23–77); sex males/females (n) – 302/96; BMI, kg/m2 (ME, range) - 28 (24-33); SAPS II at admission, grades – 12.9±3.8; pancreatic necrosis etiology, %: alcohol 62.6%, biliary 20.6%, idiopathic 12.2%, traumatic 4.6%. All patients in our study (n=398) divided randomly into three groups.

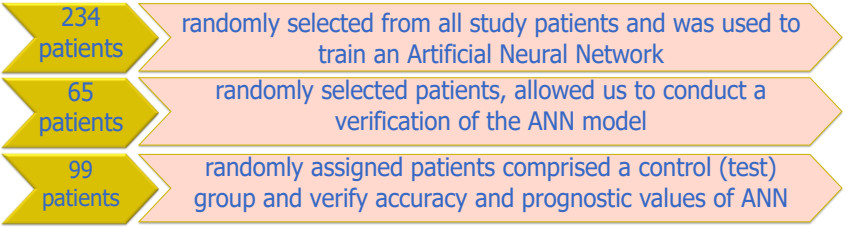


Fig. 1. Groups of patients with severe acute pancreatitis

Presenting data on admission and at 48 hours were collected: Parameters at admission: Age, sex, BMI, Ranson criteria, GS.

Consequently studied parameters: SAPS II score, multiple organ system failure, systemic inflammatory response syndrome, respiratory rate, intra-abdominal hypertension, hemoglobin, hematocrit, platelet count, prothrombin, fibrinogen, glucose, BUN, creatinine, albumin, bilirubin, ALT, AST, calcium, C-reactive protein, LDH, pO2/ fiO2, base excess et al.

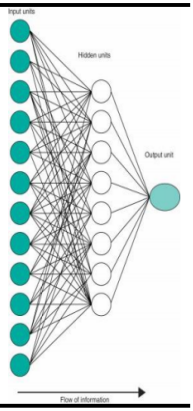
CDSS (Artificial Neural Networks) were created and trained to predict development of IPN and mortality from AP; 25% of the data set was withheld from training and was used to evaluate the accuracy of the CDSS. Accuracy of the CDSS in predicting infected pancreatic necrosis was compared with SAPS II scores. Since ANN is a computer model based on a binary number of data entry and computation, we coded clinical and laboratory data and introduced it into the system as coded values of 1 and 0. Normal laboratory results were coded as 0 and pathological results as 1. In data which didn't have normal or pathological values, such as sex or age, we used codes 0 or 1. All results (studied parameters) were coded in a binary data mode and introduced to ANN for training and data analysis. ANN design: a three-layered back-error propagation forward data flow. Back propagation of error allowed minimization of error by adjusting weights connected to ANN units, which was used in system's training to improve predicting values of the method.

3. Results

A total of 1690 patients with acute pancreatitis were identified of whom 398 (23.6%) fulfilled the clinical and radiological criteria for severe acute pancreatitis and 98 patients died (5.8%). Median SAPS II score at 48 hours was 6 (range, 0 to 23). Our CDSS with 12 criteria (see Table 1) was more accurate than SAPS II scoring systems at predicting infected pancreatic necrosis ($P<0.05$, respectively) (see Table 2). As we have demonstrated, a CDSS can be trained successfully to assist clinicians in predicting which patients are likely to have an IPN. This information could then be used to schedule FNA. The ANN that we constructed was able to differentiate between patients with sterile and infected pancreatic necrosis on the basis of the combination of a small number of clinical, laboratory, and radiologic parameters. Early diagnosis of IPN is crucial to allow the initiation of treatment, such as percutaneous or surgical drainage, appropriate antibiotic coverage, and aggressive supportive therapy.

Table 1. Input variables used to develop the artificial neural network model for IPN prediction

Variable	Data format	Error	Sensitivity (%)
Early surgery (days 0-14)*	Yes / No	0.52%	2.38%
Duration of treatment in ICU (days)	Continuous variable	0.38%	1.75%
Duration of in-hospital stay (days)	Continuous variable	0.37%	1.71%
Clinical deterioration§	Yes / No	0.36%	1.67%
Serum glucose level‡	Continuous variable	0.34%	1.56%
Temperature	Continuous variable	0.33%	1.52%
Heart rate	Continuous variable	0.25%	1.15%
Serum blood urea nitrogen	Continuous variable	0.24%	1.10%
Leukocyte count	Continuous variable	0.24%	1.08%
Respiratory rate	Continuous variable	0.23%	1.06%
CRP	Continuous variable	0.22%	1.00%
Paresis (intra-abdominal hypertension)	Yes / No	0.21%	1.00%



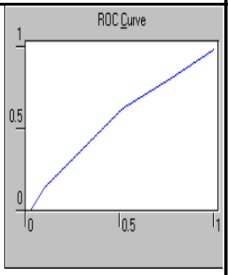
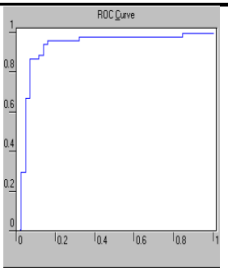
*Early surgery was conducted in some cases before IPN developed

§Clinical deterioration represented a "second wave" of the disease and was based on interpretation either APACHEII or SAPSII scores

‡Serum glucose level in non-diabetic patients

The ANN we constructed was able to determine the presence or absence of infected pancreatic necrosis based on the patients' clinical, laboratory, and radiographic parameters with an AUC=0.92. High sensitivity is important in the evaluation of IPN.

Table 2. Prediction of development of infected pancreatic necrosis using SAPS II in comparison to ANN

	SAPS II	ANN	SAPS II	ANN
Sensitivity	31.9%	41.5%		
Specificity	42.6%	96.8%		
Accuracy	53.6%	94.7%		
Positive predictive value	62.8%	92.2%		
Negative predictive value	44.1%	96.8%	AUC=0.62	AUC=0.92

4 Conclusion

Clinical decision support system was able to predict the development of infected necrotizing pancreatitis with considerable accuracy and outperformed other clinical risk scoring systems.

As computer-aided diagnostic techniques are becoming increasingly common, we can take advantage of this technology and determine which clinical questions can be successfully addressed with these tools. For many patients with severe acute pancreatitis, SAPS II or APACHE II scores are calculated daily to determine their prognosis; similar parameters could be entered into a workstation and analyzed by an ANN to evaluate cases in which patients are suspected of having an infected pancreatic necrosis. As we have demonstrated, an ANN can be trained successfully to assist clinicians in predicting which patients are likely to have an IPN. This information could then be used to schedule FNA.

References

1. Capobussi M., Banzi R., Moja L., [et al.]. Computerized decision support systems: EBM at the bedside // Recenti Prog Med. (2016) 107(11):589–591.
2. Van den Heever M., Mittal A., Haydock M., Windsor J. The use of intelligent database systems in acute pancreatitis – a systematic review // Pancreatology. (2014) 14(1): 9–16.
3. Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. Insights Imaging (2018) 9: 611.
4. Hamet P., Tremblay // J. Artificial intelligence in medicine. Metabolism. (2017) 69S:S36–S40.
5. Hashimoto D.A., Rosman, G., Rus D., Meireles O.R. Artificial Intelligence in Surgery: Promises and Perils // Ann Surg. (2018) 268(1):70–76.

М. М. АЛКЕЗУИНИ, В. И. ГОРБАЧЕНКО
Пензенский государственный университет
gorvi@mail.ru

ОБУЧЕНИЕ СЕТЕЙ РАДИАЛЬНЫХ БАЗИСНЫХ ФУНКЦИЙ ПРИ РЕШЕНИИ ЗАДАЧ АППРОКСИМАЦИИ И УРАВНЕНИЙ В ЧАСТНЫХ ПРОИЗВОДНЫХ

Рассматривается применение сетей радиальных базисных функций для бессеточных аппроксимаций функций и решения краевых задач для дифференциальных уравнений в частных производных. Предложены, реализованы и исследованы алгоритмы обучения сетей радиальных базисных функций на основе методов ускоренного градиента Нестерова и Левенберга–Марквардта.

Ключевые слова: сети радиальных базисных функций, аппроксимация функций, уравнения в частных производных, обучение, метод ускоренного градиента Нестерова, метод Левенберга–Марквардта.

Введение

В настоящее время при решении задач аппроксимации функций становятся популярными бессеточные методы [1], использующие в качестве базисных функций радиальные базисные функции (РБФ) [2]. РБФ – это функция, значение которой в некоторой точке зависит от расстояния между точкой и параметром РБФ, называемым центром. Параметры РБФ задаются, а веса находятся из условий равенства нулю невязок между аппроксимированными значениями и известными значениями функции в пробных точках. Недостатком применения РБФ является неформальный подбор параметров. Применение сетей радиальных базисных функций (СРБФ) [3] позволяет преодолеть этот недостаток.

РБФ перспективны в бессеточных методах решения дифференциальных уравнений в частных производных (ДУЧП) [4]. С применением РБФ строится приближенное аналитическое дифференцируемое решение задачи (порядок производных зависит от применяемых РБФ), что облегчает задание граничных и дополнительных условий. По сути, с помощью РБФ аппроксимируется неизвестное решение. Веса вычисляются из условия равенства нулю невязок в пробных точках, а параметры базисных функций выбираются. Возникает та же проблема подбора параметров базисных функций, которую можно решить применением СРБФ.

При использовании СРБФ решение задачи формируется в процессе обучения сети. При этом настраиваются как веса, так и параметры РБФ. Поэтому

актуальной является разработка быстрых алгоритмов обучения СРБФ. Целью данной работы является адаптация современных градиентных методов первого порядка и метода второго порядка — метода Левенберга–Марквардта для обучения СРБФ при решении задач аппроксимации функций и ДУЧП. Данная работа является развитием исследований [5].

Обучение СРБФ при решении задач аппроксимации функций

СРБФ включает два слоя [3]. Первый слой состоит из РБФ, производящих нелинейное преобразование входного вектора $\mathbf{x} = [x_1, x_2, \dots, x_d]$ – координат точки, в которой вычисляется приближение к решению, d – размерность пространства. РБФ – это функции расстояния точки пространства от параметра функции, называемого центром функции: $\varphi(\|\mathbf{x}-\mathbf{c}\|, \mathbf{p})$, где $\mathbf{x}$ – точка пространства, $\mathbf{c}$ – центр радиальной базисной функции, $\|\mathbf{x}-\mathbf{c}\|$ – евклидова норма (расстояние) между точкой и центром, $\mathbf{p}$ – вектор параметров функции. Применяются различные РБФ. В данной работе используется функция Гаусса (гауссиан) $\varphi(\|\mathbf{x}-\mathbf{c}\|, a) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{c}\|^2}{2a^2}\right)$, где $\mathbf{c}$ – положение центра функции, a – параметр формы, часто называемый шириной. Второй слой СРБФ представляет собой линейный взвешенный сумматор. Выход сети имеет вид

$$u(\mathbf{x}) = \sum_{i=1}^{n_{RBF}} w_i \varphi_i(\mathbf{x}, \mathbf{p}_i), \quad (1)$$

где n_{RBF} — количество РБФ, w_i — вес РБФ φ_i, $\mathbf{p}_i$ — вектор параметров. РБ-функции φ_i.

При решении задач аппроксимации входными векторами являются векторы координат пробных точек, а целевыми значениями — известными значениями функции в пробных точках. После обучения сеть позволяет определить значение функции в произвольной точке области определения функции. Обучение СРБФ представляет собой минимизацию функционала ошибки:

$$I = \frac{1}{2}\sum_{j=1}^{n} e_j^2 = \frac{1}{2}\sum_{j=1}^{n}\left(u(\mathbf{p}_j) - T_j\right)^2, \quad (2)$$

где n – количество пробных точек, e_j – ошибка решения в j-й пробной точке, $\mathbf{p}_j$ – координаты j-й пробной точки, $u(\mathbf{p}_j)$ – решение (1) в j-й пробной точке, T_j – целевое значение в j-й пробной точке, множитель $1/2$ введен для упрощения вычислений.

Для обучения РБФ применяются градиентные методы. В известных работах [6] используется градиентный метод первого порядка – метод градиентного спуска. Если $\boldsymbol{\Theta}$ – вектор параметров сети (в нашем случае это векторы весов, центров и ширины СРБФ), то настройка вектора $\boldsymbol{\Theta}$ на (k)-й итерации градиентного спуска описывается следующим образом:

$$\boldsymbol{\Theta}^{(k)} = \boldsymbol{\Theta}^{(k-1)} + \Delta\boldsymbol{\Theta}^{(k)}, \quad (3)$$

где $\Delta\boldsymbol{\Theta}^{(k)} = -\eta \mathbf{g}_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^{(k-1)}\right)$ – поправка вектора $\boldsymbol{\Theta}$, η – подбираемый числовой

коэффициент (скорость обучения), $\mathbf{g}_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^{(k-1)}\right)$ – вектор градиента функционала (2) по значению параметра $\boldsymbol{\Theta}$ на итерации $k-1$.

Процесс вычислений по (3) продолжается до малого значения функционала ошибки (2) или среднеквадратической погрешности.

В обучении сетей глубокой архитектуры популярны простые ускоренные градиентные методы [7]: градиентный спуск с импульсом [8] и метод ускоренного градиента Нестерова (NAG) [9]. Применение этих алгоритмов для обучения СРБФ впервые предложено в [5]. В алгоритме градиентного спуска с импульсом поправка к вектору параметров формируется следующим образом:

$$\Delta\boldsymbol{\Theta}^{(k)} = \alpha\Delta\boldsymbol{\Theta}^{(k-1)} - \eta\mathbf{g}_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^{(k-1)}\right), \quad (4)$$

где η – скорость обучения, α – коэффициент момента, принимающий значения в интервале [0, 1].

Выражение (4) содержит слагаемые, зависящие и не зависящие от градиента. На плоских участках целевой функции и вблизи локальных минимумов слагаемое, не зависящее от градиента, начинает доминировать в (4), что способствует выходу из этой области. NAG отличается от градиентного спуска с импульсом более точным вычислением поправки:

$$\Delta\boldsymbol{\Theta}^{(k)} = \alpha\Delta\boldsymbol{\Theta}^{(k-1)} - \eta\mathbf{g}_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^{(k-1)} + \alpha\Delta\boldsymbol{\Theta}^{(k-1)}\right).$$

Методы второго порядка не получили распространения при обучении СРБФ. Известны только отдельные примеры применения метода Левенберга–Марквардта в областях, не связанных с решением задач аппроксимации и ДУЧП. Для обучения СРБФ перспективным является алгоритм на основе метода доверительных областей, предложенный в [10] для решения ДУЧП и исследованный в [11–12]. Но этот метод достаточно сложен и требует на каждой итерации обучения решать задачу условной минимизации.

Рассмотрим применение для обучения СРБФ метода Левенберга–Марквардта [13] при аппроксимации функций двух переменных. Введем единый вектор параметров сети:

$$\boldsymbol{\theta} = \left[w_1, \ldots, w_{n_{RBF}}, c_{11}, \ldots, c_{n_{RBF}1}, c_{12}, \ldots, c_{n_{RBF}2}, a_1, \ldots, a_{n_{RBF}}\right]^T,$$

где w_j – вес, c_{j1} и c_{j1} – координаты центра, a_j – ширина параметры j-й РБ-функции ($j = 1,2,3, \ldots, M$).

Коррекция вектора параметров $\boldsymbol{\theta}$ в k-м цикле (итерации) обучения производится по формуле $\boldsymbol{\theta}^{(k)} = \boldsymbol{\theta}^{(k-1)} + \Delta\boldsymbol{\theta}^{(k)}$, где вектор поправки $\Delta\boldsymbol{\theta}^{(k)}$ находится из решения системы линейных алгебраических уравнений

$$\left(\mathbf{J}_k^T\mathbf{J}_k + \mu_k\mathbf{E}\right)\Delta\boldsymbol{\theta}^{(k)} = -\mathbf{g}_k, \quad (5)$$

где $\mathbf{E}$ – единичная матрица, μ_k – параметр регуляризации, изменяющийся на каждом шаге обучения, $\mathbf{g} = \mathbf{J}^T\mathbf{e}$ – вектор градиента функционала (2), $\mathbf{e} = [e_1, e_2, \ldots, e_{N+K}]^T$ – вектор ошибок сети, $\mathbf{J}_k$ – матрица Якоби, вычисленная по значениям параметров сети в k –1 итерации.

Матрица Якоби имеет блочный вид $\mathbf{J}=\left[\mathbf{J}_{\mathrm{w}}\ \mathbf{J}_{\mathrm{c}_1}\ \mathbf{J}_{\mathrm{c}_2}\ \mathbf{J}_{\mathrm{a}}\right]$. Так как у СРБФ только два слоя и из них один линейный, то матрица $\mathbf{J}$ может быть вычислена аналитически. Так, элементы матрицы $\mathbf{J}_{\mathrm{w}}$ с учетом (2) и (1) имеют вид $\frac{\partial \mathrm{e_i}}{\partial \mathrm{w_j}} = \frac{\partial}{\partial \mathrm{w_j}}[\mathrm{u}(\mathbf{p}_\mathrm{i}) - \mathrm{T_i}] = \varphi_\mathrm{j}(\mathbf{p}_\mathrm{i})$, где $\varphi_\mathrm{j}(\mathbf{p}_\mathrm{i})$ – значение j-й РБФ в пробной точке $\mathbf{p}_\mathrm{i}$. Элементы матриц $\mathbf{J}_{\mathrm{c}_1}$ и $\mathbf{J}_{\mathrm{c}_2}$ имеют вид $\frac{\partial \mathrm{e_i}}{\partial \mathrm{c_{j1}}} = \mathrm{w_j}\varphi_\mathrm{j}(\mathbf{p}_\mathrm{i})\frac{\mathrm{p_{i1}}-\mathrm{c_{j1}}}{\mathrm{a_j^2}}$, $\frac{\partial \mathrm{e_i}}{\partial \mathrm{c_{j2}}} = \mathrm{w_j}\varphi_\mathrm{j}(\mathbf{p}_\mathrm{i})\frac{\mathrm{p_{i2}}-\mathrm{c_{j2}}}{\mathrm{a_j^2}}$. Элементы матрицы $\mathbf{J}_{\mathrm{a}}$ имеют вид $\frac{\partial \mathrm{e_i}}{\partial \mathrm{a_j}} = \mathrm{w_j}\varphi_\mathrm{j}(\mathbf{p}_\mathrm{i})\frac{\|\mathbf{p}_\mathrm{i}-\mathbf{c}_\mathrm{j}\|^2}{\mathrm{a_j^3}}$.

В процессе обучения параметр регуляризации μ должен изменяться. Процесс начинается при относительно большом значении параметра. Это означает, что в начале процесса обучения в (5) $\mathbf{J}_\mathrm{k}^\mathrm{T}\mathbf{J}_\mathrm{k} + \mu_\mathrm{k}\mathbf{E} \approx \mu_\mathrm{k}\mathbf{E}$ и реализуется метод градиентного спуска с малым шагом. По мере уменьшения функционала ошибки параметр μ уменьшается и метод приближается к методу Ньютона с аппроксимацией Гессиана $\mathbf{H} \approx \mathbf{J}_\mathrm{k}^\mathrm{T}\mathbf{J}_\mathrm{k}$, что обеспечивает хорошую скорость сходимости вблизи минимума функционала ошибки. Рекомендуется начинать с некоторого значения μ_0 и коэффициента $\nu > 1$ [14]. Текущее значение μ делится на ν, если функционал ошибки уменьшается или умножается на ν, если функционал ошибки увеличивается.

В [13] показано, что метод Левенберга–Марквардта эквивалентен методу доверительных областей, с радиусом доверительной области, регулируемым параметром μ. Но в отличие от известных реализаций метода доверительных областей метод Левенберга–Марквардта проще, так как не требует решения на каждой итерации обучения достаточно сложной задачи условной оптимизации.

Недостатком метода Левенберга–Марквардта является плохая обусловленность системы (5), зависящая от начальных значений ширины РБФ и увеличивающаяся с ростом точности обучения сети [5]. Для функции Гаусса с ростом ширины значения РБФ в $\mathbf{J}_{\mathrm{w}}$ стремятся к единице, а элементы матриц $\mathbf{J}_{\mathrm{c}_1}$, $\mathbf{J}_{\mathrm{c}_2}$ и $\mathbf{J}_{\mathrm{a}}$ стремятся к нулю, обусловленность матрицы $\mathbf{J}$ ухудшается. В пределе матрица становится особенной. Ухудшению обусловленности способствует трансформация Гаусса $\mathbf{J}_\mathrm{k}^\mathrm{T}\mathbf{J}_\mathrm{k}$. Параметр регуляризации улучшает обусловленность, но уменьшение параметра μ по мере уменьшения погрешности обучения приводит к ухудшению обусловленности. Имеются проблемы с выбором параметра регуляризации: малое значение параметра приводит к низкой скорости сходимости, большое приводит к негладкому характеру уменьшения погрешности.

Обучение СРБФ при решении ДУЧП

Решение ДУЧП на RBFN рассмотрим на примере задачи [10]:

$$\mathrm{Lu}(\mathbf{x}) = \mathrm{f}(\mathbf{x}), \mathbf{x} \in \Omega; \mathrm{Bu}(\mathbf{x}) = \mathrm{p}(\mathbf{x}), \mathbf{x} \in \partial\Omega, \quad (6)$$

где u – искомое решение; L – дифференциальный оператор, например, $\mathrm{L} = \frac{\partial^2}{\partial \mathrm{x}^2} + \frac{\partial^2}{\partial \mathrm{y}^2}$; B – оператор граничных условий; Ω – область решения; $\partial\Omega$ — граница области; f и p – известные функции.

Решение (6) основано на минимизации функционала ошибки — суммы квадратов невязок в пробных точках внутри и на границе области решения

$$\mathrm{I}(\mathbf{w}, \mathbf{p}) = \sum_{\mathrm{i}=1}^{\mathrm{N}} [\mathrm{Lu}_{\mathrm{RBF}}(\mathbf{x}_\mathrm{i}; \mathbf{w}, \mathbf{p}) - \mathrm{f}(\mathbf{x}_\mathrm{i})]^2 + \lambda \sum_{\mathrm{i}=\mathrm{N}+1}^{\mathrm{N}+\mathrm{K}} [\mathrm{Bu}_{\mathrm{RBF}}(\mathbf{x}_\mathrm{i}; \mathbf{w}, \mathbf{p}) - \mathrm{p}(\mathbf{x}_\mathrm{i})]^2, \quad (7)$$

где $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_\mathrm{N} \in \Omega, \mathbf{x}_{\mathrm{N}+1}, \mathbf{x}_{\mathrm{N}+2}, \ldots, \mathbf{x}_{\mathrm{N}+\mathrm{K}} \in \partial\Omega$; λ – подбираемый множитель, учитывающий вклад граничных пробных точек, $\mathrm{u}_{\mathrm{RBF}}$ – приближение к решению (1), полученное РБФ-сетью, N, K – количество пробных точек внутри и на границе области решения соответственно.

В качестве примера рассмотрим модельную задачу, описываемую уравнением Лапласа с граничными условием Дирихле:

$$\frac{\partial^2 \mathrm{u}}{\partial \mathrm{x}_1^2} + \frac{\partial^2 \mathrm{u}}{\partial \mathrm{x}_{21}^2} = \mathrm{f}(\mathrm{x}_1, \mathrm{x}_2), \ (\mathrm{x}_1, \mathrm{x}_2) \in \Omega, \ \mathrm{u} = \mathrm{p}(\mathrm{x}_1, \mathrm{x}_2) \in \partial\Omega, \quad (8)$$

где $\partial\Omega$ – граница области; f и p – известные функции $(\mathrm{x}_1, \mathrm{x}_2)$.

Для данного примера функционал ошибки (7) имеет вид

$$\mathrm{I} = \sum_{\mathrm{i}=1}^{\mathrm{N}} (\Delta \mathrm{u}_\mathrm{i} - \mathrm{f}_\mathrm{i})^2 + \lambda \sum_{\mathrm{j}=1}^{\mathrm{K}} \left(\mathrm{u}_\mathrm{j} - \mathrm{p}_\mathrm{j}\right)^2, \quad (9)$$

где $\Delta \mathrm{u}_\mathrm{i}$ – значение Лапласиана в точке i.

Компоненты градиента по весам и параметрам РБФ несложно вычислить аналитически, например,

$$\frac{\partial \mathrm{I}}{\partial \mathrm{w}_\mathrm{p}} = \sum_{\mathrm{i}=1}^{\mathrm{N}} (\Delta \mathrm{u}_\mathrm{i} - \mathrm{f}_\mathrm{i}) \mathrm{e}^{-\frac{\|\mathbf{x}_\mathbf{i} - \mathbf{c}_\mathbf{p}\|^2}{2\mathrm{a}_\mathrm{p}^2}} \frac{\|\mathbf{x}_\mathbf{i} - \mathbf{c}_\mathbf{p}\|^2 - 2\mathrm{a}_\mathrm{p}^2}{\mathrm{a}_\mathrm{p}^4} + \lambda \sum_{\mathrm{j}=1}^{\mathrm{K}} \left(\mathrm{u}_\mathrm{j} - \mathrm{p}_\mathrm{j}\right) \mathrm{e}^{\frac{\|\mathbf{x}_\mathbf{i} - \mathbf{c}_\mathbf{p}\|^2}{2\mathrm{a}_\mathrm{p}^2}}.$$

Процесс обучения сети заканчивается при малом значении функционала ошибки (9) или средней квадратической погрешности.

Рассмотрим адаптацию метода Левенберга–Марквардта для обучения СРБФ на примере задачи (8). Фигурирующими в матрице Якоби ошибками являются невязки в пробных точках. Для СРБФ элементы матрицы Якоби несложно вычислить аналитически. Например, элементы матрицы $\boldsymbol{J}_w$ для внутренних пробных точек вычисляются по формуле $\frac{\partial e_i}{\partial w_j} = \frac{\partial \Delta u_i}{\partial w_j} = \varphi_j(\boldsymbol{x}_i) \frac{\|\boldsymbol{x}_i - \boldsymbol{c}_j\|^2 - 2a_j^2}{a_j^4}$, где $\varphi_j(\boldsymbol{x}_i)$ – значение j-й РБФ в пробной точке $\boldsymbol{x}_i$. Для граничных пробных точек вычисления производятся по формуле $\frac{\partial e_i}{\partial w_j} = \varphi_j(\boldsymbol{x}_i)$.

Условия завершения процесса обучения методом Левенберга–Марквардта те же, что и в методе Нестерова.

Экспериментальное исследование

Эксперименты проводились в системе MATLAB. Для решения системы (5) использовался решатель систем линейных алгебраических уравнений MATLAB. В [5] на примере аппроксимации несложной функции $z = x^2 + y^2$ показано, что NAG для средней квадратической ошибки, равной 0.01, на порядок уменьшает число итераций по сравнению с градиентным спуском. Метод Левенберга–Марквардта более чем на порядок сокращает число итераций по сравнению с NAG. Но методы первого порядка практически не позволили обучать сеть до меньшей средней квадратической ошибки, в то время как метод Левенберга–Марквардта позволил в среднем за 40 итераций получить среднюю квадратическую погрешность, равную 10^{-6}.

Было проведено тестирование предложенных алгоритмов с использованием популярной в аппроксимации функции Франке [15] (рис. 1):

$$f(x,y) = 0{,}75exp\left(-\frac{(9x-2)^2}{4} - \frac{(9y-2)^2}{4}\right) + 0{,}75exp\left(-\frac{(9x+1)^2}{49} - \frac{9y+1}{10}\right) + 0{,}5exp\left(-\frac{(9x-7)^2}{4} - \frac{(9y-3)^2}{4}\right) - 0{,}2exp(-(9x-4)^2 - (9y-7)^2).$$

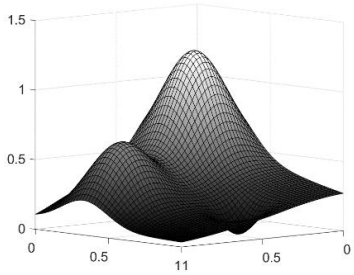
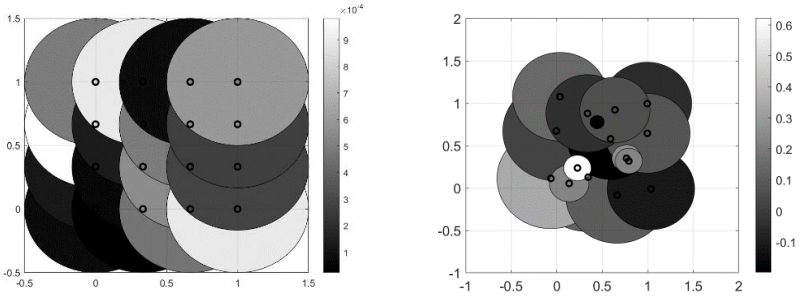


Рис. 1. График функции Франке

Методы первого порядка при аппроксимации функции Франке не позволили получить среднюю квадратическую погрешность менее 10^{-1}. Метод Левенберга–Марквардта в среднем за 20 итераций позволил достичь средней квадратической погрешности, равной 10^{-6}. Аппроксимация проводилась в области $(x = 0 \cdots 1, y = 0 \cdots 1)$. Узлы интерполяции располагались случайным образом в области. Количество нейронов равно 16. В начальном состоянии центры РБФ располагались на сетке (рис. 2а). Веса инициировались случайными числами, равномерно распределенными от 0 до 0,001. Начальная ширина всех РБФ равно 5. Лучшие результаты получены при $\mu = 2{,}0$ и $\nu = 1{,}5$.

На рис. 2 показаны расположения центров, условное обозначение ширины (в виде окружностей с радиусами, равными ширине) и весов (в виде заливки окружностей) РБФ перед обучением и после обучения сети. Рис. 2 показывает радикальное изменение координат центров и ширины в процессе обучения сети, что является обоснованием важности настройки центров и ширины, а не только весов.



а) перед обучением сети б) после обучения сети

Рис. 2. Центры и ширина РБФ при решении задачи аппроксимации

Эксперименты по решению ДУЧП проводилось на примере решения задачи (8) при $f(x_1, x_1) = sin(\pi x_1)\, sin(\pi x_1)$ и $p(x_1, x_1) = 0$. Задача решалась в единичном квадрате. Количество внутренних пробных точек $N = 100$. Количество граничных пробных точек $K = 40$. Штрафной коэффициент функционала ошибки по границе равен $\lambda = 10$. При инициализации центры РБФ располагались на квадратной сетке 8×8. Пробные точки располагались случайным образом в области решения и на границе области. Веса инициировались нулевыми значениями. Начальная ширина всех РБФ была постоянной, равной 0,2 для всех методов. Зависимость среднеквадратической ошибки различных методов от номера итерации показана на рис. 3.

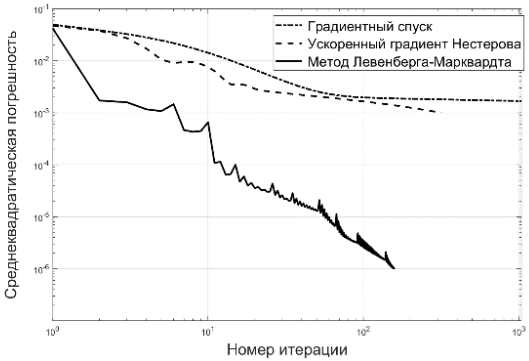
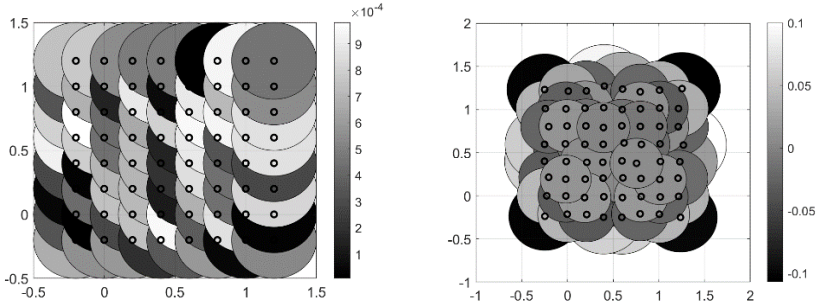


Рис. 3. Зависимость среднеквадратической ошибки от номера итерации

Расположение центров и условное представление ширины перед обучением сети (рис. 4а) и после обучения (рис. 4б) показывают важность настройки параметров RBF.



а) перед обучением сети б) после обучения сети

Рис. 4. Центры и ширина РБФ при решении ДУЧП

Метод градиентного спуска позволил решить модельную задачу с небольшой точностью. Для решения с большой точностью метод практически неприменим. Несколько бóльшую точность обеспечивает метод Нестерова. Только метод Левенберга–Марквардта позволил решить задачу с высокой точностью за приемлемое время. Метод Левенберга–Марквардта показал соизмеримые результаты по сравнению с методом доверительных областей [10], но реализация метода Левенберга–Марквардта проще. Недостатками метода Левенберга-Марквардта являются плохая обусловленность системы (5), формирующей коррекцию параметров, и негладкий характер сходимости.

Заключение

Разработаны и исследованы ускоренные алгоритмы первого порядка и алгоритм на основе метода Левенберга–Марквардта для решения задач аппроксимации функций и решения краевых задач на сетях радиальных базисных функций. Предложенные алгоритмы показали свою эффективность.

Дальнейшие исследования будут включать в себя расширение классов решаемых задач и дальнейшее исследование методов обучения второго порядка, в частности, исследование возможности использования методов, не требующих вычисления Гессиана (Hessian-Free).

Список литературы

1. Fasshauer G.F. Meshfree Approximation Methods with MATLAB. World Scientific Publishing Company. 2007. 520 p.

2. Buhmann M.D. Radial Basis Functions: Theory and Implementations. Cambridge University Press. 2009. 272 p.
3. Aggarwal C.C. Neural Networks and Deep Learning. Springer, 2018. 497 p.
4. Chen W., Fu Z.-J. Recent Advances in Radial Basis Function Collocation Methods. Springer. 2014. 90 p.
5. Алкезуини М.М., Горбаченко В.И. Совершенствование алгоритмов обучения сетей радиальных базисных функций для решения задач аппроксимации // Модели, системы, сети в экономике, технике, природе и обществе. 2017. № 3 (23). С.123–138.
6. Yadav N., Yadav A., Kumar M. An Introduction to Neural Network Methods for Differential Equations. Springer. 2015. 115 p.
7. Гудфеллоу Я., Бенджио Т., Курвиль А. Глубокое обучение. Москва : ДМК Пресс. 2018. 625 с.
8. Polyak B.T. Some methods of speeding up the convergence of iteration methods // USSR Computational Mathematics and Mathematical Physics. 1964. Vol. 4, No 5. P. 1–17.
9. Sutskever I., Martens J., Dahl G., Hinton G. On the importance of initialization and momentum in deep learning // ICML'13 Proceedings of the 30th International Conference on International Conference on Machine Learning. 2013. Vol. 28. P. III-1139-III-1147.
10. Горбаченко В.И., Жуков М.В. Решение краевых задач математической физики с помощью сетей радиальных базисных функций // Журнал вычислительной математики и математической физики. 2017. Т. 57, № 1. С. 133–143.
11. Alqezweeni M.M., Gorbachenko V.I., Zhukov M.V., Jaafar M.S. Efficient Solving of Boundary Value Problems Using Radial Basis Function Networks Learned by Trust Region Method // International Journal of Mathematics and Mathematical Sciences. Vol. 2018, Article ID 9457578. 4 pages. 2018.
12. Елисов Л.Н., Горбаченко В.И., Жуков М.В. Обучение методом доверительных областей сетей радиальных базисных функций при решении краевых задач // Автоматика и телемеханика. 2018. № 9. С. 95–105.
13. Marquardt D.W. An Algorithm for Least-Squares Estimation of Nonlinear Parameters // Journal of the Society for Industrial and Applied Mathematics. 1963. Vol. 11. No 2. P. 431–441.
14. Gill P.E., Murrey W., Wright M. Practical Optimization. Academic Press. 1982. 418 p.
15. Franke R. Scattered data Interpolation: Tests of some Methods // Mathematics of Computation. 1982. Vol. 38. No 157. P.181–200.

Д. И. СОРОКИН[1], **А. С. НУЖНЫЙ**[2],
Е. А. САВЕЛЬЕВА-ТРОФИМОВА[2]
[1] Московский физико-технический институт
(национальный исследовательский университет)
[2] Институт проблем безопасного развития атомной энергетики РАН
dmitrii.sorokin@phystech.edu

ПРОГРАММА ИЕРАРХИЧЕСКОГО ПОИСКА В КОРПУСЕ ТЕКСТОВОЙ ДОКУМЕНТАЦИИ ПО ВОПРОСАМ ЗАХОРОНЕНИЯ РАДИОАКТИВНЫХ ОТХОДОВ

Представлена программа иерархического поиска в текстовой документации. Программа представляет корпус документов в виде карты семантически близких кластеров. Для каждого кластера автоматически вычисляются ключевые слова и на основании их выделяются темы. Благодаря этому пользователь может искать фрагменты текстов, относящиеся к определенным темам. Тематический анализ текстового корпуса позволяет обнаруживать наличие или отсутствие в нем тех или иных тем, оценивать полноту содержащейся информации.

Ключевые слова: обработка естественного языка, семантический анализ, кластеризация, машинное обучение.

Введение

В ИБРАЭ РАН накоплен большой объем текстовой документации по объектам захоронения радиоактивных отходов (РАО) [1]. Общее количество различных документов на данный момент превышает 6000. Тематический анализ «вручную» часто не представляется возможным, поэтому требуется привлечение методов обработки естественного языка.

В работе описывается программа, строящая иерархическую рубрикацию текстового массива данных, группируя тексты в рубрики и подрубрики. При этом для каждой рубрики автоматически формируется краткое описание, состоящее из ключевых слов, наилучшим образом отображающих ее содержание. Такое описание в дальнейшем служит пользователю ориентиром для поиска по иерархии тематик. Спускаясь от крупномасштабных тем к более детальным, пользователь, в конечном итоге, находит интересующую его текстовую информацию узконаправленного содержания.

Опираясь на построенную тематическую пирамиду, программа может осуществлять тематический поиск в массиве документов, запросом в котором служит достаточно крупный фрагмент текста или документ. По сути, такой поиск близок к идее Exploration search [2]. Для предоставленного текста программа определяет тематики и находит тематически наиболее близкие к

нему документы из обучающей выборки.

Кроме того, программа позволяет оценивать полноту собранной информации по конкретным объектам захоронения радиоактивных отходов и выявлять недостаточно раскрытые темы. В работе также представлен использованный при разработке алгоритмов математический аппарат и результаты апробации программы.

Предобработка PDF файлов

Анализируемые документы представлены в формате PDF. Файлы, созданные в разные годы и полученные разными путями (набранные в TeX или конвертированные из различных версий программы Microsoft Word), имеют в PDF различное описание. Это существенно усложняет задачу построения алгоритмов их обработки – разбиения текста на фрагменты (главы, абзацы и т.п.) и удаления из рассмотрения служебной информации (колонтитулов, оглавлений и т.п.). В языке Python существует несколько широко используемых пакетов для работы с файлами PDF формата:

- Pdfmier [3];
- PyPDF2 [4];
- Poopler [5].

Нами был выбран пакет Poopler, как предоставляющий лучшую скорость и качество обработки файлов, имеющихся в хранилище.

Поскольку операция считывания и обработки PDF занимает много времени, было решено препроцессинг корпуса документов проводить автономно, а его результат – считанный, очищенный от служебной информации и разбитый на абзацы текст, сохранять в базу данных формата sqlite (https://www.sqlite.org). Кроме самого текста в базу данных сохраняется указание имени файла, из которого он взят, и номера страниц.

При тематическом поиске фрагменты текста часто рассматриваются в приближении «мешка слов», когда учитывается только факт присутствия слова в документе, но не берется в рассмотрение их порядок. При анализе текстов в приближении «мешка слов» теряют смысл падежи и склонения. В этом случае целесообразно привести все слова к так называемой нормальной форме: существительные – к именительному падежу, единственному числу; прилагательные – к именительному падежу единственному числу, мужскому роду; глаголы, причастия, деепричастия – к нормальной форме глагола. Нормализация слов русского языка позволяет сжать словарь корпуса примерно в 5 раз, что, в конечном итоге, увеличивает точность поиска и рубрикации. Для нормализации слов в данной программе использовалась библиотека pymorphy [6].

Следующие шаги, практически всегда применяемые при анализе текстов и также призванные повысить точность поиска – это удаления из рассмотре-

ния так называемых стоп-слов (союзов, предлогов и других часто употребляемых слов) и отбрасывание редко встречающихся слов. Последние часто являются опечатками и вносят шум в данные.

Для тематического поиска часто используются методы, основанные на векторном представлении слов – когда каждому слову ставится в соответствие вектор в признаковом пространстве. Такие векторные представления стремятся создать описание, при котором близкие по смыслу слова, синонимы, имеют близкие по евклидову расстоянию векторы, далекие, антонимы, – далекие. В этом случае всему тексту можно поставить в соответствие вектор, рассчитанный как средний вектор входящих в него слов. В случае удачного построения векторного представления векторы тематически близких текстов будут близки, далеких – далеки.

Программа использует заранее построенный словарь соответствия слова вектору для расчета вектора фрагмента текста. Эта процедура также ресурсоемкая, поэтому было решено расчет среднего вектора проводить автономно, а результат сохранять в базу данных.

В результате обработки корпуса в базе данных будут сохранены фрагменты текста, с указанием исходного документа и номера страницы, их векторное представление и обратный индекс, рассчитанные после нормализации слов в текстах.

Векторное представление слов

Для отображения слов в векторное пространство используется подход word2vec [7]. Метод заключается в обучении нейронной сети, состоящей из входного, скрытого и выходного слоев предсказывать по слову (или их последовательности) в тексте следующее слово. В word2vec существует две основные архитектуры: COBOW и Skip-gram. В архитектуре COBOW нейронная сеть обучается предсказывать текущее слово, получая на вход окружающий его контекст, порядок слов контекста при этом не учитывается. В Skip-gram нейронная сеть обучается предсказывать контекст исходя из текущего слова. В результате обучения нейросети фиксируются ее синаптические коэффициенты. При этом коэффициенты ее первого слоя в дальнейшем могут быть использованы как векторы, расстояния между которыми отражают семантическую близость слов.

Для построения отображения слов в векторное представление программа использует архитектуру COBOW из библиотеки GenSim (https://radimrehurek.com/gensim).

Иерархическая кластеризация

Для кластеризации программа строит карту Кохонена на множестве векторов фрагментов текста [8]. Карта Кохонена – метод кластеризации данных, при котором кластеры упорядочены между собой в виде гексагональной или

прямоугольной сетки. Процедуру кластеризации данных с помощью самоорганизующихся карт можно описать как подстройку двумерной сетки кластеров под рельеф данных в многомерном пространстве. В результате карта даст аппроксимацию некоторого двумерного подмножества в многомерном признаковом пространстве, включающего в себя или близко лежащего к точкам обучающей выборки.

Математически карта Кохонена задается множеством центров кластеров – точек в признаковом пространстве $\{m_k\}_{k=1}^{K}$, где K – их число. В качестве меры смысловой близости фрагмента текста к рассматриваемой тематике берется расстояние до ближайшего кластера карты:

$$r = \min_k \|x - m_k\| \quad (1)$$

Затем кластеры карты Кохонена кластеризуются алгоритмом k-means на 5 и 12 кластеров. При меньшем числе кластеров полученные темы будут иметь более общий смысл, в то время как при большем числе кластеров темы будут иметь более частный смысл.

Для каждого кластера вычисляются ключевые слова в соответствии с метрикой:

$$score(word) = in\_cluster(word)^2 / total(word), \quad (2)$$

где $in\_cluster(word)$ – число вхождений слова в текущий кластер, а $total(word)$ – суммарное число слов. Таким образом, слова, уникальные для данного параграфа, будут иметь большое значение метрики, а слова, встречающиеся во многих фрагментах текстов, – маленькое.

Тематический поиск осуществляется на основании выделенных ключевых слов алгоритмом Okapi BM25 [9].

Описание программы

Программа представляет собой клиент-серверное приложение, написанное на Flask.

Программа

- выполняет предобработку корпуса PDF документов, в которую входит построение обратного индекса и расчет векторного представления текстов;
- выполняет построение карты Кохнена и кластеризацию;
- вычисляет ключевые слова для каждого кластера;
- ранжирует документы по близости к каждому кластеру.

На рис. 1 и рис. 2 показан интерфейс программы. В левой части находится карта Кохонена с выделенными на ней кластерами. Каждый кластер соответствует некоторой автоматически выделенной теме. При наведении курсора мыши на кластер он подсвечивается и выводятся ключевые слова, отнесенные к этому кластеру. В правой части находится поисковая выдача. При

клике на кластер в поисковой выдаче показывается список ссылок на документы и тексты фрагментов, отнесенные к данному кластеру. При клике на ссылку выбранный документ открывается в отдельном окне на нужной странице.

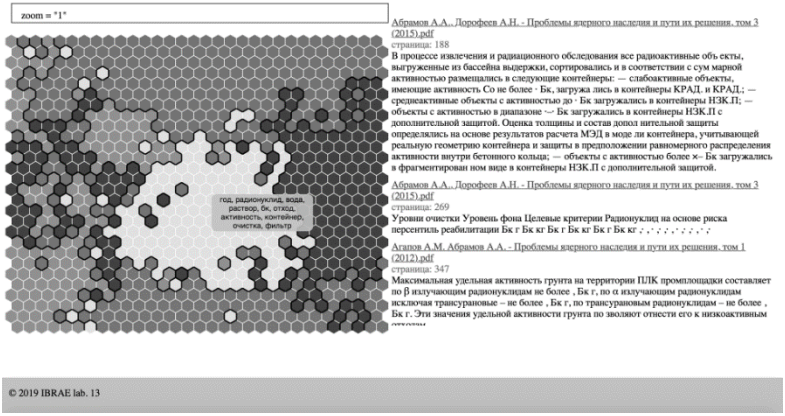


Рис.1. Окно программы иерархического поиска текстовой документации (zoom = 1)

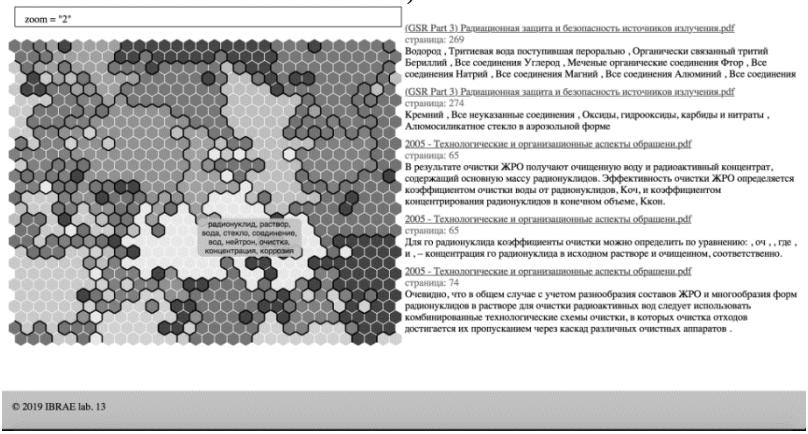


Рис.2. Окно программы иерархического поиска текстовой документации (zoom = 2)

Результаты тестирования программы

Для тестирования результатов работы автоматического поиска тем в документах сравним ключевые слова, выделенные для родительского кластера (при zoom = 1) и дочерних кластеров (при zoom = 2).

Кластер «безопасность, программа, рао, эксплуатация, требование, работа, радиоактивный отход, должный, организация, обращение» содержит в себе:

- «программа, эксплуатация, работа, должный, рао, проект, безопасность, требование, организация, вывод»;
- «радиоактивный отход, атомный энергия, федеральный закон, российский федерация, магатэ, статья, безопасность, государственный, норма, федеральный»;
- «отход, рао, оят, хранение, захоронение, переработка, хранилище, сао, нао, тро»;
- «облучение, доза, риска, человек, эффективный доза, население, мзв, ткань, уровень, ситуация».

Кластер «модель, параметр, значение, метод, моделирование, оценка, процесс, исследование, порода, уравнение» содержит в себе:

- «скважина, зона, геологический, участок, район, километр, тектонический, разлом, порода, массив»;
- «модель, метод, параметр, моделирование, функция, анализ, неопределённость, задача, расчёт, процесс»;
- «напряжение, шпур, коэффициент, температура, рис, деформация, уравнение, прочность, значение, энергия».

Видно соответствие тем родительских и дочерних кластеров.

Заключение

В статье описана программа, выполняющая иерархический поиск текстовой информации. Фрагменты текстов ранжируются по соответствию автоматически выделенным темам. Тестирование автоматического выделения тем, ключевых слов и поиска фрагментов текстов по ним продемонстрировало точность, указывающую на практическую целесообразность его применения.

Список литературы

1. Сорокин Д.И., Нужный А.С. Разработка нейросетевой системы анализа специализированной документации // Нейроинформатика-2018. XX Всероссийская научно-техническая конференция. Сборник научных трудов. Ч. 2. Москва : МИФИ. 2018. С. 10-19.
2. White R.W., Roth R.A. Exploratory Search: Beyond the Query-Response Paradigm. Synthesis Lectures on Information Concepts, Retrieval, and Services // Morgan and Claypool Publishers. 2009.
3. http://pdfminer-docs.readthedocs.io
4. https://pythonhosted.org/PyPDF2
5. https://poppler.freedesktop.org
6. Korobov M. Morphological Analyzer and Generator for Russian and Ukrainian Languages // Analysis of Images, Social Networks and Texts. 2015. P. 320–332 .

7. Mikolov T., Chen K., Corrado G., Dean J. Efficient Estimation of Word Representations in Vector Space // Proceedings of Workshop at ICLR. 2013.
8. Kohonen T. Self-Organizing maps. Springer Berlin Heidelberg. 2001.
9. Robertson S.E., Walker S., Hancock-Beaulieu M. Okapi at TREC-7 // Proceedings of the Seventh Text REtrieval Conference. Gaithersburg, USA, November 1998.

НЕЙРОБИОЛОГИЯ И НЕЙРОБИОНИКА

И.В. НУЙДЕЛЬ[1,3], А.В. КОЛОСОВ[1,3], С.А. ПОЛЕВАЯ[1,2], В.Г. ЯХНО[1,3]

[1]Нижегородский государственный университет им. Н.И. Лобачевского
[2]Приволжский исследовательский медицинский университет
[3]Институт прикладной физики РАН, Нижний Новгород
s453383@mail.ru

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ДИНАМИКИ АЛЬФА-РИТМА ЭЭГ ПРИ РИТМИЧЕСКОЙ ФОТОСТИМУЛЯЦИИ В ПРОЦЕССЕ НЕЙРОБИОУПРАВЛЕНИЯ*

В работе представлены нейроинформационные решения проблемы персонализации нейробиоуправления [1]. Вычислительные эксперименты показали эффективность феноменологической математической модели элементарной таламокортикальной ячейки для описания динамики альфа-ритма ЭЭГ при ритмической фотостимуляции в процессе нейробиоуправления. Дано обоснование возможности применения модели в качестве адаптивного симулятора индивидуальных ритмических паттернов ЭЭГ, обеспечивающего генерацию персонализированных протоколов нейробиоуправления для оптимизации функций мозга.

Ключевые слова: модель таламо-кортикальной системы, нейроинтерфейс, электроэнцефалограмма (ЭЭГ), нейробиоуправление, ритмы мозга.

Введение

Задачей сегодняшнего дня является персонализированный подход к медицинской диагностике, исследованиям и оказанию медицинской помощи [2]. Для развития этого актуального направления необходимо развитие симуляторов сложных систем, позволяющих наглядно продемонстрировать наличие взаимных связей, прямых и обратных, между диагностической информацией, например сигналами электроэнцефалограммы (ЭЭГ) или электрокардиограммы (ЭКГ), и информационными маркерами состояния обследуемого человека (например, объективными, полученными в результате измерений, и субъективными данными, полученными в результате опроса, его психофи-

* Работа выполнена при частичной поддержке Российского фонда фундаментальных исследований (гранты РФФИ № 18-013-01225 А, 19-013-00095 А, 18-413-520006 р-а) и Министерства образования и науки РФ (проект №14.Y26.31.0022).

зического состояния). Технологии «интерфейс мозг–компьютер» [3] и нейробиоуправление направлены для достижения именно этой цели. В основе технологий лежат методы модуляции активности мозга с помощью сигналов обратной связи от биопотенциалов мозга, в частности от собственных ритмических процессов человека, а именно его электроэнцефалограммы (ЭЭГ), что и является нейробиоуправлением [4, 5]. В работе продемонстрированы возможности математической модели элементарной таламокортикальной ячейки для описания частотно-временных откликов мозга на ритмическое аудиовизуальное воздействие. Задача настоящей работы – обработка и воспроизведение эффектов динамики сигнала ЭЭГ, которые наблюдаются в ходе биоуправления; разработка примера симулятора сложной системы преобразования сигналов человеком.

Как преобразуется сигнал в обрабатывающей информацию таламокортикальной системе? Что произойдет, если в систему подать частотно-модулированный сигнал с линейно увеличивающейся частотой?

В данной работе проведены вычислительные эксперименты по моделированию ситуации, когда внешний сигнал с частотными характеристиками входного сигнала управляет выходным сигналом системы. Это модель биоуправления [1].

Описание нейронных модулей модели

В экспериментальных нейрофизиологических исследованиях выявлено, что взаимосвязанные нейрональные модули: кора, ретикулярные ядра таламуса, специфический таламус – играют важную роль в процессах обработки информации.

Взаимодействующие звенья таламокортикальной цепи определяют архитектуру феноменологической модели таламокортикальной ячейки.

В реальности структура одного нейронного модуля состоит из ансамблей пирамидных нейронов и тормозных интернейронов коры (Cortex), нейронов специфических таламических ядер (специфические ядра, Thalamus) и тормозных нейронов ретикулярного ядра таламуса (неспецифические ядра Thalamus reticular nucleus, NRT), связанных между собой.

Схема межмодульного взаимодействия показана на рис.1 [6]. Треугольниками на схеме показаны возбудительные, а кружками – тормозные связи между модулями. Стрелка в нижней части рисунка – сенсорный вход в таламус.

Таламус – сложное полифункциональное образование, включающее релейные ядра**,** где переключается афферентация от органов чувств в соответствующие области коры больших полушарий. То есть в специфических (сенсорных, или релейных) ядрах таламуса происходит синаптическое переключение сенсорной информации с аксонов восходящих афферентных путей на следующие, конечные нейроны, отростки которых идут в соответствующие

сенсорные проекционные области коры больших полушарий. Например, специфическим ядром зрительной сенсорной системы является латеральное коленчатое тело (ЛКТ, LGN), имеющее прямые связи с затылочными (зрительными) проекционными областями коры больших полушарий [6].

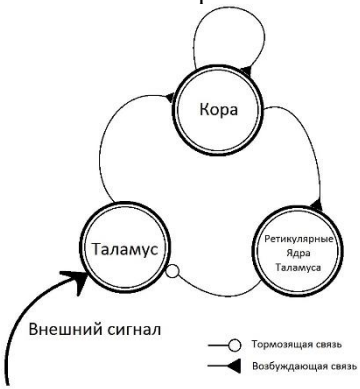


Рис. 1. Схема функциональных связей между подсистемами в таламокортикальной ячейке

В нормальных условиях функционирования в таламокортикальной системе происходит следующее: 1) Внешний сигнал активирует вначале нейроны специфического (релейного) ядра таламуса 2) По таламокортикальному пути возбуждение поступает в кору, причем корковые пирамиды связаны с тормозными интернейронами, которые могут определенным образом модулировать их активность. 3) Дальнейшее распространение возбуждения происходит по нисходящим корково-таламическим путям к специфическому и ретикулярному ядрам таламуса. 4) Последнее связано с релейным ядром таламуса тормозными связями и может прерывать поступление возбуждения из специфического таламического ядра в кору.

Ретикулярное ядро таламуса является своеобразными воротами для сенсорной информации, поступающей в кору. Оно не имеет прямого выхода на кору. Вместе с тем оно получает входы от коры и других ядер таламуса и, по-видимому, выполняет функцию внутриталамического регулятора.

Тормозное влияние ретикулярных ядер таламуса заканчивает «строб» активности, во время которого, например, происходит выделение признака исходного сенсорного сигнала в коре головного мозга человека и животных; за ним следует период неактивности.

В результате после прохождения входного сигнала через таламокортикальную систему на коре формируется прерывистое, стробированное представление исходного сенсорного сигнала.

В работе рассматривается сосредоточенная модель функционирования таламокортикальной системы без учета строения внутренних структурных

модулей системы: таламическое специфическое ядро; тормозное ретикулярное неспецифическое ядро таламуса, кора, связанная с ядрами таламуса [7].

Так как все внешние входы коры связаны с таламическими структурами, на модели проведены расчёты сигналов и их спектров в случае частотной модуляции таламического сигнала внешним сигналом с линейно возрастающей частотой.

Входной сенсорный сигнал является частотно-модулированным сигналом (в психофизическом эксперименте он аналогичен инфракрасному высокочастотному модулированному сигналу по отношению к собственной частоте сигнала ЭЭГ). Сигнал от переменной коры аналогичен интегральному биоэлектрическому сигналу ЭЭГ.

Протокол нейробиоуправления

Резонансное нейробиоуправление с двойной обратной связью на базе ПАК «BioFeedBack2» проводится по гибридному протоколу: Фон – до/после: 2 мин. запись фоновой вертексной ЭЭГ: активный электрод – Cz, заземляющий и референтный электроды на мочках ушей; сканирование по частоте - 210 с: воздействие импульсным инфракрасным излучением с сальтаторно нарастающей частотой от 8 до 14 Гц, шаг по частоте – 0.1 Гц, шаг по времени – 3 с и музыкоподобным звуковым сигналом, тональность и громкость которого определяется пиковой амплитудой в спектре текущей ЭЭГ в диапазоне 8–14 Гц. Характеристическое время обратной связи – 10 мс, точность по частоте 0,2–0,4 Гц. В звуковой сигнал добавлены периодические шумовые импульсы, предъявляемые с частотой, соответствующей фоновой ЧСС [1, 5].

В ходе эксперимента мерцающая (с линейно увеличивающейся частотой) инфракрасная лампа была направлена на закрытые глаза испытуемого. ЭЭГ испытуемого снимается до воздействия на него импульсным инфракрасным излучением, в процессе воздействия и после воздействия. Далее производится построение динамического спектра сигнала.

В норме динамический спектр таламокортикальной ЭЭГ человека имеет вид, показанный на рис. 2. справа. Различиями для каждого отдельного испытуемого являются лишь основная собственная частота альфа-ритма в диапазоне от 8 до 12 Гц и ширина спектрального размытия вокруг неё.

Рис. 2. показывает, что, находясь под внешним воздействием, таламокортикальная система испытуемого работает в режиме вынужденных колебаний в течение определенного интервала времени. Этот интервал появляется в момент, когда частота внешнего сигнала выше собственной частоты таламокортикальной системы и существует до того, как частота внешнего сигнала становится настолько большой, что испытуемый воспринимает сигнал как постоянный с амплитудой, равной средней величине периодического сигнала. Этот динамический режим соответствует вынужденным колебаниям заполнения с автоколебательной огибающей и очень хорошо виден на ЭЭГ при «наложении» на неё формы внешнего сигнала.

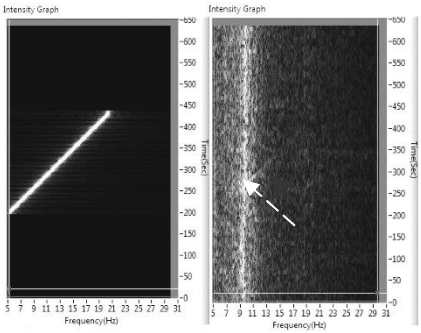


Рис. 2. Динамические спектры для внешнего инфракрасного сигнала с линейно возрастающей частотой (слева) и для соответствующей таламокортикальной ЭЭГ одного из испытуемых (справа). Частоты отложены на горизонтальной оси, время отложено на вертикальной оси

Результаты

Для расчетов использована ранее разработанная феноменологическая модель элементарной таламокортикальной ячейки [7], включающая в себя взаимодействующие модули, соответствующие таким нейронным модулям мозга, как таламус, кора и ретикулярные ядра таламуса.

Феноменологическая модель элементарной таламокортикальной ячейки, соответствующая схеме на рис. 1, описывается системой дифференциальных уравнений (1) – (3) для моделирования обработки внешнего сигнала между таламусом, корой и ретикулярными ядрами таламуса:

$$\frac{dU_1}{dt} = -\frac{U_1}{\tau_1} + k_1 \cdot F_1\left[-T_1 + k_{вх}U_{вх} + k_{13}U_{13}\right], \quad (1)$$

$$\frac{dU_2}{dt} = -\frac{U_2}{\tau_2} + k_2 \cdot F_2\left[-T_2 + k_{21}U_1 + k_{22}U_2\right], \quad (2)$$

$$\frac{dU_3}{dt} = -\frac{U_3}{\tau_3} + k_3 \cdot F_3\left[-T_3 + k_{32}U_2\right], \quad (3)$$

где $U_1; U_2; U_3$ – усредненная активность нейронов выбранных участков таламуса, коры и ретикулярных ядер таламуса соответственно; τ_i – характерное время затухания активности в соответствующих нейронных ансамблях; k_i – амплитуда генерации импульсной активности соответствующими нейронными ансамблями; T_i – усредненные величины для порогов возбуж-

дения соответствующих нейронных ансамблей; $U_{\text{вх}}$ – входной сигнал, поступающий на таламус; k_{ij} – коэффициенты взаимной связи между подсистемами в таламо-кортикальной ячейке; $F_i[\ ]$ – ступенчато-образные функции, крутизна которых характеризует разброс величин порогов около усредненных значений в рассматриваемом ансамбле (другое название: функции энергообеспечения), в квадратных скобках – аналог постсинаптического потенциала (ПСП) на мембранах соответствующих ансамблей нейронов.

В данной работе из всех параметров менялась только величина внешнего сигнала $U_{\text{вх}}$. Численные значения параметров: $k_i = 1,\ i = 1,2,3;$ $\tau_i = 1,\ i = 1,2,3;\ T_1 = 0;\ T_2 = 0.5;\ T_3 = 0.5;\ k_{ex} = 1;\ k_{13} = 1;\ k_{21} = 1;\ k_{22} = 0;\ k_{32} = 1.$ Использовалась плавно меняющаяся функция энергообеспечения

$$F_i\left[0.5 + 0.5 \cdot \tanh\left(20 \cdot [\ ]\right)\right].$$

Для $U_{\text{вх}}$ был взят сигнал, представленный на рис. 3а, который аналогичен экспериментальному: частота осциллирующей части линейно возрастает.

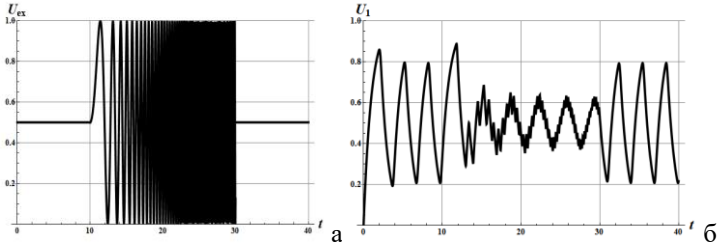


Рис. 3. а – временная форма внешнего сигнала в модели; б – отклик таламуса на внешний сигнал с линейно возрастающей частотой

Отклик таламуса (U_1) для такого типа внешнего сигнала проиллюстрирован на рис. 3б. Рис. 3 демонстрирует три состояния системы: перед внешним воздействием, в процессе внешнего высокочастотного воздействия и после воздействия. Все состояния характеризуются собственной частотой таламокортикальной системы, которая зависит от средней величины внешнего сигнала [7].

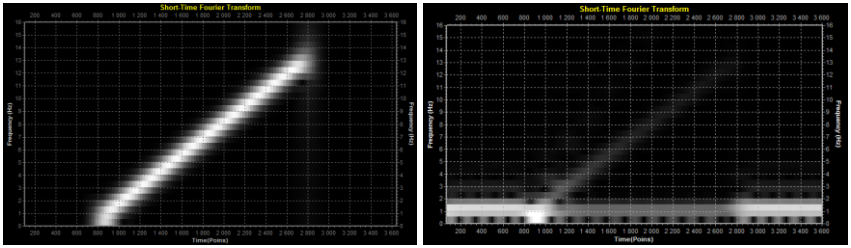


Рис. 4. а – динамический спектр внешнего сигнала ($U_{\text{вх}}$); б – динамический спектр таламического сигнала (U_1). Частоты отложены на вертикальной оси, время отложено на горизонтальной оси

«Высокочастотное воздействие» означает, что частота внешнего сигнала существенно больше собственной частоты таламокортикальной ячейки. Кроме того, приведем динамические спектры для обоих (внешнего и таламического) сигналов, которые изображены на рис. 5а и рис. 5б соответственно.

Выводы

Целью данной работы являлось обоснование применения феноменологической математической модели элементарной таламокортикальной ячейки для описания частотно-временных откликов реальной таламокортикальной системы, т.е. различных модуляций альфа-ритма. Так как частотное поведение реального сигнала ЭЭГ и поведение модельного сигнала сходны, представляется возможным предсказывать изменение базового альфа-ритма внешним воздействием, для которого известны параметры таламокортикальной системы. Эта информация позволит совершенствовать существующие процедуры биоуправления с обратной связью, способствующие активизации познавательной деятельности человека. Путём проведения комплекса тренировок удаётся управлять частотой альфа-ритма (нейробиоуправление) таким образом, что у испытуемых по объективным показателям психофизиологической диагностики происходит усиление когнитивной деятельности, а, по их субъективным оценкам, улучшается самочувствие в целом [8, 9].

Для дальнейших исследований в психофизическом эксперименте и на модели интересно рассмотреть реализацию «противоположного» динамического режима: вынужденных колебаний огибающей таламического отклика и автоколебаний заполнения под воздействием внешнего сигнала низкой частоты по сравнению с собственной частотой таламокортикальной системы.

В работе продемонстрирован нейроинформационный подход к персонализированному управлению ритмами мозга. На модели проиллюстрирован персонализированный подход к медицинской диагностике, исследованиям и оказанию медицинской помощи: на феноменологической модели таламокортикальной ячейки удается воспроизводить индивидуальные особенности сложной системы обработки информации.

Авторы выражают благодарность А.А. Тельных (ИПФ РАН, Нижний Новгород) за программное обеспечение Signal-Spectrum View, которое было использовано для построения динамических спектров сигналов.

Список литературы

1. Федотчев А.И., О Сан Чжун, Бондарь А.Т., Семёнов В.С., Современные возможности и подходы к активизации когнитивной деятельности и процессов обучения у человека. Монография. Пущино: ИБК РАН, 2017. 114 с.
2. Hammond D. C. (2011) What is Neurofeedback: An Update, Journal of Neurotherapy, 15:4,305-336, doi: 10.1080/10874208.2011.623090
3. Miranda R.A., Casebeer W.D., Hein A.M., Judy J.W., Krotkov E.P., Laabs T.L., Manzo J.E., Pankratz K.G., Pratt G.A., Sanchez J.C. Darpa-funded efforts in the development of novel brain–computer interface technologies. J. Neurosci. Methods. 2015; 244:52–67. doi: 10.1016/j.jneumeth.2014.07.019.
4. Федотчев А.И., Парин С.Б., Полевая С.А., Великова С.Д. Технологии «интерфейс мозг-компьютер» и нейробиоуправление: современное состояние и перспективы клинического применения. Современные технологии в медицине. 2017; 9(1): 175-184. doi: 10.17691/stm2017.9.1.01.
5. Федотчев А.И., Бондарь А.Т., Бахчина А.В., Парин С.Б., Полевая С.А., Радченко Г.С. Музыкально-акустические воздействия, управляемые биопотенциалами мозга, в коррекции неблагоприятных функциональных состояний // Успехи физиологических наук. 2016; 47(1): 67-79.
6. Coulter D. A. (1997). Thalamocortical Anatomy and Physiology. In Engel, Jr., J. and Pedley, T.A., editors, Epilepsy: A Comprehensive Textbook, pages 341-351, Philadelphia. Lippincott-Raven.
7. Колосов А. В., Нуйдель И. В., Яхно В. Г. Исследование динамических режимов в математической модели элементарной таламокортикальной ячейки // Известия вузов. Прикладная нелинейная динамика. 2016 Т. 24, вып. 5. С. 72-83. doi: 10.18500/0869-6632-2016-24-5-72-83
8. Bondar A., Shubina L. Nonlinear reactions of limbic structure electrical activity in response to rhythmical photostimulation in guinea pigs. Brain Research Bulletin, V. 143, October 2018, Pages 73-82. doi:10.1016/j.brainresbull.2018.10.002
9. Федотчев А.И., Бондарь А.Т., Бахчина А.В., Григорьева В.Н., Катаев А.А., Парин С.Б., Полевая С.А., Радченко Г.С. Трансформация ЭЭГ осцилляторов пациента в музыкоподобные сигналы при коррекции стресс-индуцированных функциональных состояний // Современные технологии в медицине 2016; 8(1): 93-98. doi: 10.17691/stm2016.8.1.12.

И. В. МАКУШЕВИЧ, Н. Г. БИБИКОВ
Акустический институт им. Н.Н. Андреева
nbibikov1@yandex.ru

ОПЫТ ПРИМЕНЕНИЯ ИНДЕКСА ХЕРСТА ДЛЯ ИЗУЧЕНИЯ ДИНАМИКИ СПОНТАННОЙ АКТИВНОСТИ НЕЙРОНОВ СЛУХОВОЙ СИСТЕМЫ*

Головной мозг является динамической системой, находящейся в непрерывном изменении с элементами самоподобия во временной области. Его изучение естественно осуществлять методами фрактального анализа. В данной работе для анализа спонтанной активности нейронов слуховой системы травяной лягушки применен подход, основанный на динамическом анализе вариабельности временного точечного процесса с использованием индекса Херста. Он позволил показать, что спонтанная активность изменяет свою вариабельность, переходя от случайных колебаний к трендовой зависимости, и наоборот.

Ключевые слова: спонтанная активность, фрактальность, индекс Херста, слуховая система, амфибии.

Введение

Современный этап исследования мозга характеризуется повышенным интересом к функционированию ансамблей или кластеров нейронных элементов. Новые методики картирования мозга, включающие кальциевое высвечивание и разнообразные варианты динамического мониторинга функционирования нейронных ансамблей, основанные на геномной модификации клеток, явно продемонстрировали неустойчивость поведения нейронных структур. Постоянно возникают новые аттракторы, приводящие к изменению связей между участками мозга и формированию тесно взаимодействующих между собой кластеров взаимосвязанных клеток.

Одним из проявлений этой особенности работы мозга является существование медленных и сверхмедленных изменений электрического потенциала фиксированных точек сенсорных структур мозга, обнаруженное и исследуемое уже в течение длительного времени [1]. Исследование процессов, демонстрирующих и проявляющих эту особенность нейронных структур, естественно осуществлять методами фрактального анализа.

Несколько методик фрактального анализа в последние годы довольно широко используются и при анализе импульсной активности одиночных

*Данная работа выполнена при частичной поддержке РФФИ, проект № 16-04-01006-а.

нейронов. Чаще всего проводят исследование зависимости значения фактора Фано (отношение дисперсии числа спайков к его среднему значению) от величины анализируемого временного промежутка. Степенную зависимость этой функции с положительным показателем обычно интерпретируют как проявление фрактальных свойств исследуемого точечного временного процесса [2]. Данным методом показана фрактальность процессов импульсной активности нейронов во многих центрах головного мозга животных. Наряду с исследованием указанной зависимости в нескольких публикациях анализировали такие же процессы, используя фактор Аллана. Он позволяет несколько более подробно охарактеризовать процесс, классифицируя его либо как хаотический шум, либо как броуновское движение [3].

В некоторых областях науки и практики для изучения процессов, характеризующихся хаотичностью и неустойчивым возникновением определенных тенденций, довольно широко используют подход, впервые предложенный гидрологом Херстом для описания сезонных ежегодных колебаний уровня Нила [4]. Подход основан на довольно простой методике подсчета максимального и минимального значений анализируемого параметра. Для чисто случайного процесса это значение должно расти пропорционально корню квадратному из величины интервала анализа. Для хаотических и фрактальных процессов значение индекса Херста обычно превышает это значение. Использование для анализа спонтанной активности нейронов вычисления фрактального индекса Херста было начато нами в 2017 г. и уже отражено в нескольких работах [5–7].

В работе [5] был проведен сравнительный анализ подходов Фано, Аллана и Херста для изучения спонтанной активности слуховых нейронов продолговатого мозга травяной лягушки. Показана довольно тесная связь индекса Херста с более широко используемыми при анализе нейронной активности индексами Фано и Аллана и принципиальная перспективность использования индекса Херста при исследовании активности клеток слуховой системы. При этом была сформулирована задача применения данного подхода к динамическому анализу спонтанной импульсации во времени. Именно эта проблема рассматривается в данной работе.

Методика

Исходные записи последовательности спайков были зарегистрированы у нейронов дорсального ядра продолговатого мозга и полукружного торуса среднего мозга обездвиженных травяных лягушек в отсутствие контролируемых внешних сигналов. Эти два ядра составляют основной прямой слуховой путь, в котором обрабатывается информация о звуковых стимулах. Методика подготовки объекта и регистрации активности нейронных элементов с помощью стандартных микроэлектродов многократно описана. Исходные записи хранились в компьютере в виде последовательности межимпульсных

интервалов. Эта информация и использовалась нами для последовательного вычисления динамики индекса Херста.

Наблюдение за временной динамикой изменения индекса Херста позволяет выявить существование тренда в исследуемом процессе. Если временной точечный процесс имеет тенденцию либо к постоянному уменьшению, либо к постоянному увеличению интервалов, то разность между максимальным и минимальным значениями данного параметра растет быстрее, чем у чисто случайного процесса. Если же параметр характеризуется повторением определенных фрагментов, то рост этой разности замедляется или может вовсе прекратиться. Наиболее широко данный подход применяется при анализе различных финансовых проблем, прежде всего, с целью выявления скрытых тенденций котировки биржевых индексов и курсов обмена валют. При наличии тренда изучаемого параметра безотносительно к его направлению значение индекса Херста должно быть существенно выше 0,5.

Существует несколько вариантов реальной динамической оценки индекса Херста. В настоящей работе используется вариант, предложенный экономистом Найманом [6]. В его работе введен модифицированный индекс Херста, и на основании численных расчетов предложена таблица пороговых значений, относительно которых временной участок процесса, согласно значению индекса Херста, относится к антитрендовым, случайным или трендовым.

Исходная формула для вычисления индекса, предложенная Херстом [5]:

$$H = \frac{log\left(\frac{R}{S}\right)}{log\left(\frac{1}{2}N\right)},$$

где N – число измерений (членов временного ряда),
S – среднеквадратичное отклонение ряда наблюдений,
R – размах накопленного отклонения:

$$R = max_{1 \le u \le N}(Z_u) - min_{1 \le u \le N}(Z_u),$$

где Z_u – значения межимпульсных интервалов.

Найман внес некоторые изменения в стандартный индекс Херста с целью совершенствования расчета для коротких временных рядов. Предложенные им модифицированные формулы для вычисления индекса Херста следующие:

$$H_T = \frac{log\frac{R}{S_T}}{log\frac{\pi}{2}N} \cdot (-0.0011 \cdot ln(N) + 1.0136),$$

где $\frac{R}{S_T} = \frac{R}{S} \cdot 0.998752 + 1.051037.$

Найманом же были предложены критерии отнесения участка процесса к трендовым, антитрендовым и случайным. Они зависят от объема выборки.

Мы вычисляли индекс Херста для полного ряда межимпульсных интервалов «пошагово», т.е. первый подряд включал участок сигнала от первого

до N-го интервала, второй от второго до N + 1 и далее до того подряда, конечное значение которого соответствовало последнему межспайковому интервалу. Использовали значения N в 2000, 3000, 4000 и 5000 межимпульсных интервалов при условии, что число спайков в записи хотя бы в два раза превосходило N.

Результаты

Всего исследована спонтанная активность 91 клетки продолговатого мозга и 48 клеток слухового центра среднего мозга травяной лягушки. Анализировались процессы с общим числом межимпульсных интервалов от 1270 до 36786.

Значительное большинство нейронов обоих исследованных центров характеризовалось медленными и хаотичными изменениями динамики процесса от случайного к трендовому и наоборот. Достоверных участков антитрендовой активности обнаружено не было. Полный анализ полученной информации будет осуществлен в отдельной публикации, а в данной работе мы ограничимся двумя примерами, иллюстрирующими характерное поведение спонтанной импульсации нейронов исследованных отделов слуховой системы.

На рис. 1 иллюстрируется динамика изменения индекса Херста для фоновой импульсации типичного нейрона дорсального ядра – основного слухового центра продолговатого мозга. Нейрон 0120006 реагировал на тональные отрезки при характеристической частоте 650 Гц и пороге в районе 50 дБ УЗД. На оси абсцисс откладывается время первого спайка в «подряде», а по оси ординат – значение индекса Херста, вычисленное для данного «подряда». Видно, что на начальном участке полной записи поведение нейрона изменяется от трендового к более случайному, затем в течение определенного времени оно изменяется около порогового значения между случайным и трендовым, но в конце вновь проявляется трендовая зависимость.

Важно отметить, что указанные существенные изменения статистических свойств спонтанной активности осуществляются в условиях отсутствия заметного общего тренда средней частоты импульсации, что иллюстрирует график, приведенный на вкладке.

На рис. 2 иллюстрируется динамика изменения индекса Херста для нейрона 0211003 полукружного торуса – слухового центра среднего мозга, в котором осуществляется выделение временных признаков поступающей звуковой информации. По своим параметрам реакции на звук эта клетка была близка к нейрону продолговатого мозга, иллюстрированному на рис. 1, выделялась среди других исследованных нейронов торуса очень высокой частотой фоновой импульсации. Характеристическая частота составляла 0,65 кГц, а порог 52 дБ УЗД. Весьма близкими являлись и значения средней частоты импульсации в отсутствии сигнала. У данного нейрона значения ин-

декса Херста также всегда превышали значение 0,5 и также наблюдался переход от близкого к случайного поведения к трендовому без явного изменения плотности импульсации. По сравнению с нейроном продолговатого мозга может быть отмечена только несколько большая плавность этих переходов.

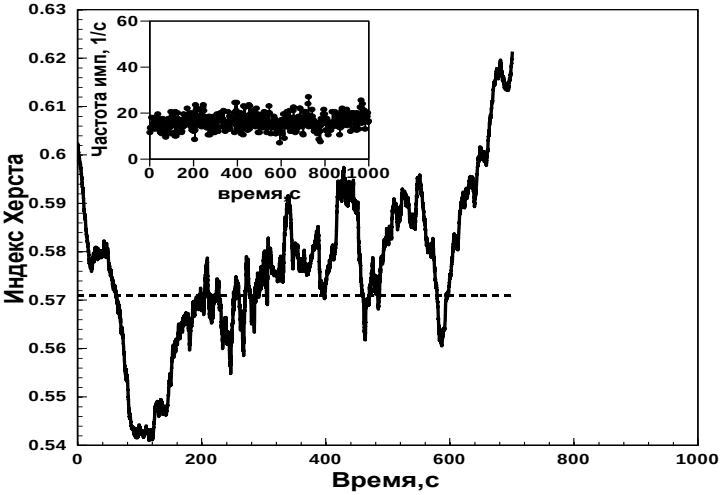


Рис. 1. Динамика индекса Херста для нейрона 0120006 слухового центра продолговатого мозга лягушки. На оси ординат – время начала «подряда», включающего по 5000 межимпульсных интервалов. Общее число интервалов – 16055. Длительность регистрации – 1000 с. Пунктирная горизонтальная линия соответствует пороговому значению для достоверной оценки процесса как трендового. На вкладке приведена зависимость частоты импульсации этого нейрона от времени

Обсуждение и выводы

В данной работе к анализу импульсной активности нейронов слуховой системы мы применили метод анализа длительных процессов, представленных последовательностью чисел, в данном случае значениями последовательных интервалов между спайками одиночного нейрона. Предполагалось, что эта последовательность, будучи хаотической, может обладать определенной памятью на прошедшие события. Метод анализа подобных процессов, предложенный уже весьма давно [4], был затем модифицирован [8] с целью оценки и даже прогнозирования финансовых потоков. Представляется, что он вполне приложим и к изучению хаотической импульсной активности, характерной для многих нейронов головного мозга, и, в том числе, тех, которые располагаются в ядрах слухового пути позвоночных. Об этом, в частности, свидетельствует неплохая корреляция индекса Херста, рассчитанного по

всей длительности реализации, с индексами Фано и Аллана, показанная нами ранее [5].

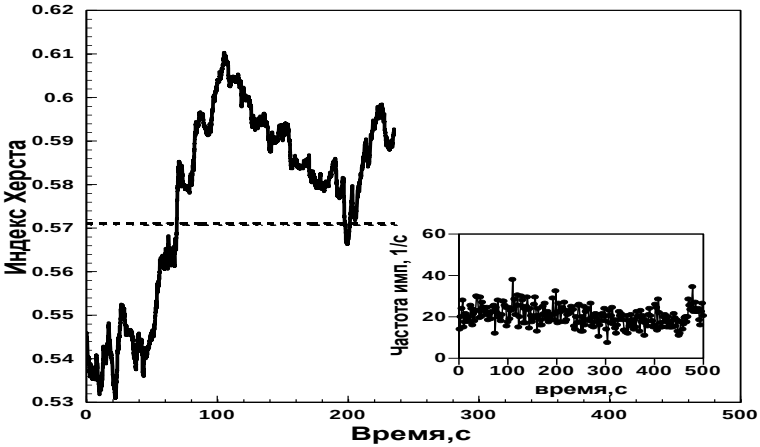


Рис. 2. Динамика индекса Херста для нейрона 0211003 полукружного торуса с высокой частотой спонтанной импульсации. На оси ординат – время начала «подряда», включающего по 5000 межимпульсных интервалов. Общее число интервалов – 10221. Длительность регистрации – 500 с. На вкладке приведена зависимость частоты импульсации этого нейрона от времени. Остальные обозначения, как на рис.1

Описанное нами в данной работе явление, которое можно назвать вариабельностью вариабельности, сводится к изменению внутренних свойств спонтанной активности при сохранении средней частоты импульсации практически неизменной.

В настоящее время довольно большое значение придается именно вариабельности нейронной импульсации. В значительном большинстве случаев ее оценивают по значению индекса Фано. В ряде работ [9,10] было показано, что значение индекса Фано обычно существенно уменьшается при воздействии стимула. Иногда уменьшение вариабельности процесса фоновой импульсации удавалось заметить при таких уровнях сигнала, при которых достоверного увеличения средней частоты импульсации еще не выявлялось.

Значение вариабельности может резко уменьшиться при анестезии и зависит от ряда других особенностей состояния животного [10]. Поэтому спонтанные изменения статистических характеристик импульсации нейрона в отсутствие контролируемых внешних воздействий представляются довольно естественными. Для более детального изучения этого эффекта требуется максимально возможная длительность регистрации при условии сравнительно высокой частоты импульсации. Следует также проанализировать связь динамики изменения индекса Херста с ходом зависимостей факторов Фано и Аллана при наибольших возможных значениях интервала анализа.

Представляется вполне естественным сопоставить обнаруженные эффекты медленного изменения вариабельности процессов спонтанной импульсации с медленными изменениями постоянных потенциалов различных структур мозга. При этом следует заметить, что все подобные медленные изменения могут происходить локально, касаясь только одной субпопуляции клеток. Весьма вероятно, что с помощью использованного нами метода – последовательного вычисления индекса Херста – для длительных записей спонтанной активности выявляются медленные и сверхмедленные изменения свойств исследуемого процесса. Относительно возможной функциональной значимости явления вариабельности вариабельности могут существовать различные гипотезы. Все они находятся в русле существующих в настоящее время представлений о кластерной организации всей нервной системы позвоночных.

Список литературы

1. Филиппов И.В., Кребс А.А., Пугачев К.С. Сверхмедленная биоэлектрическая активность структур слуховой системы головного мозга // Сенсорные системы 2006. Т. 20. №3. С. 238–244.
2. Lowen S.B., Teich M.C. Estimation and simulation of fractal stochastic point processes // Fractals. 1995. V. 3, N 1. P. 183–210.
3. Gebber G.L., Orer H.S., Barman S.M. Fractal noises and motions in time series of presympathetic and sympathetic neural activities // J. Neurophysiol. 2006. V. 95, N. 2. P. 1176–1184.
4. Hurst H.E. Long term storage capacity of reservoirs // Transactions of the American Society of Civil Engineers. 1951. V. 116. N. 1. P. 770–799.
5. Bibikov N.G., Makushevich I.V. Comparison of some fractal analysis methods for studying the spontaneous activity in medullar auditory units //in Advances in Neural Computation, Machine Learning, and Cognitive Research (Neuroinformatics 2017. Studies in Computational Intelligence. V. 736) / Ed. by B. Kryzhanovsky, W. Dunin-Barkowski, and V. Redko (Springer, Cham, 2018). P. 180–185.
6. Бибиков Н.Г., Макушевич И.В., Низамов С.В. Некоторые особенности спонтанной импульсации слуховых нейронов головного мозга амфибий // Нейрокомпьютеры: разработка, применение. 2018. № 3. С. 55–61.
7. Бибиков Н.Г., Макушевич И.В., Дымов А.Б. Фрактальные особенности фоновой импульсной активности нейронов слухового центра среднего мозга лягушки // Биофизика. 2019. Т. 64, № 3. С. 515–526.
8. Найман Э. Расчет показателя Херста с целью выявления трендовости (персистентности) финансовых рынков и макроэкономических показателей // Економіст. 2009. № 10. С.18-28.
9. Churchland M.M., Yu B.M., Cunningham J.P., Sugrue L.P., [at al.]. Stimulus onset quenches neural variability: a widespread cortical phenomenon// Nature Neuroscience. 2010. V. 13, №3. P. 369–378.
10. White B., Abbott L.F., Fiser J. Suppression of cortical neural variability is stimulus- and state-dependent // J. Neurophysiol. 2012.108: 2383–2392.

С. В. СТАСЕНКО, И. А. ЛАЗАРЕВИЧ, В. Б. КАЗАНЦЕВ
Нижегородский государственный университет им. Н.И. Лобачевского
stasenko@neuro.nnov.ru

ГЕНЕРАЦИЯ КВАЗИСИНХРОННЫХ ПАЧЕЧНЫХ РАЗРЯДОВ В МОДЕЛИ НЕЙРОН-ГЛИАЛЬНОЙ СЕТИ

В статье рассматривается эффект влияния активности глиальных клеток (астроцитов) на синаптическую динамику межнейронных контактов и сетевую динамику. Модель представляет собой сеть синаптически связанных импульсных нейронов, где динамика синаптических связей модулируется воздействием глиапередатчика. В сети синаптически связанных нейронов обнаружен эффект спонтанных переходов в режим генерации квазисинхронных пачечных разрядов, типичных для эппилептиформной активности нейронных сетей мозга.

Ключевые слова: трехчастный синапс, сетевая синхронизация, астроцит, мультистабильность, нейронная сеть.

Введение

Исследование принципов и построение адекватных моделей и обработки информации в мозге по-прежнему остается одной из ключевых задач современной нейроинформатики. С точки зрения нейробиологии основным механизмом информационного транспорта в мозге считается синаптическая передача между нейронами, которая, посредством биохимических механизмов, обеспечивает направленную передачу биоэлектрических сигналов от пре- к постсинаптическому нейрону. Однако экспериментальные исследования последних лет показали, что не только нейроны играют ключевую роль в процессе обработки информации, но и другие типы клеток. Одним из таких типов клеток являются астроциты. Астроциты – это глиальные клетки, которые могут формировать двунаправленный путь регуляции нейронной активности в мозге [1–4]. Влияние астроцитов на синаптическую передачу привело к формированию гипотезы так называемого трехчастного синапса [3]. В рамках данной гипотезы предполагается, что при синаптической передаче происходит активация астроцитов за счет связывания на рецепторах мембраны нейротрансмиттера, высвобождаемого из пресинаптического нейрона. Это в свою очередь приводит к высвобождению из астроцита нейроактивных веществ, регулирующих вероятность выброса нейротрансмиттера пресинаптическим нейроном, формируя таким образом петлю обратной связи [3].

В данной работе мы рассматриваем влияние такой обратной связи на синаптическую передачу и формирование мультистабильности, которая, как

оказалось, на сетевом уровне может приводить к спонтанному возникновению пачечных квазисинхронных биоэлектрических разрядов.

В нейровычислительных задачах принято использовать нейронные сети с множественными устойчивыми состояниями активности. В частности, реверберирующие состояния кортикальной активности принято считать основной важных когнитивных процессов и функций. Например, было продемонстрировано, что аттракторные сети в коре головного мозга определяют процессы формирования долговременной памяти [3], кратковременной памяти [4] и другие когнитивные функции [5]. Интересным проявлением бистабильности является возникновение селективной стационарной активности в префронтальной коре, то есть длительного повышения частоты колебаний нейронов в ответ на специфичные стимулы [6]. Временное распределение нейронных импульсов в периоды такой активности обладает высокой нерегулярностью, когда коэффициент вариации межимпульсных интервалов превышает единицу [7]. В качестве источника такой нерегулярности предполагается синаптическая мультистабильность, где каждому устойчивому состоянию синаптической активности соответствует определенное значение коэффициента вариации межимпульсных интервалов.

В данной работе мы показываем возможность возникновения бистабильной динамики в модели трехчастного синапса. На уровне сетевой динамики такая бистабильность приводит к формированию квазисинхронных пачечных разрядов в нейрон-глиальной сети.

Модель трехчастного синапса

Рассмотрим механизм астроцитарной регуляции синаптической передачи в рамках концепции трехчастного синапса (пресинаптический нейрон-астроцит-постсинаптичсекий нейрон). В случае одиночного синапса основной переменной, описывающей динамику системы, является концентрация нейротрансмиттера (глутамата) в синаптической щели. Эта переменная является стохастической, так как везикулярный выброс нейротрансмиттера является случайным процессом, при этом вероятность выброса может зависеть от ряда факторов – например, от присутствия астроцитарных модуляторов. Для одиночного синапса распределение приходящих на пресинаптический нейрон электрических импульсов можно аппроксимировать распределением Пуассона с характерной частотой генерации импульсов в сети. При этом вводится дополнительный параметр – вероятность выброса в случае прихода импульса на пресинаптический нейрон. Эта переменная может зависеть как и от времен предыдущих выбросов (кратковременная пластичность), так и от концентрации нейротрансмиттера. В линейном приближении динамику концентрации нейротрансмиттера можно описать следующим уравнением:

$$\frac{dX}{dt} = -\alpha_x X + k_{pre}\beta_x H_x, \qquad H_x(v_j) = \frac{1}{1+\exp(-(v_j-\theta_x)/k_x)}, \qquad (1)$$

где α_x – релаксационная константа, $\beta_x H_x$ – часть трансмиттера, продиффундировавшая к астроциту в результате выброса в щель, $H_x(v_i)$ – активационная функция, описывающая профиль концентрации нейротрансмиттера в щели в зависимости от пресинаптического мембранного потенциала, k_{pre} – стохастическая переменная, описывающая случайность везикулярного выброса.

Соответственно, динамику глиатрасмиттеров, выбрасываемых астроцитом, можно описать следующим уравнением:

$$\frac{dY_k}{dt} = -a_k Y_k + \beta_k H_k(X_e), \quad H_k(X_e) = \frac{1}{1+\exp(-\frac{(X-\theta_k)}{k_k})}. \qquad (2)$$

Здесь индекс k соответствует одному из множества выбрасываемых глиатрансмиттеров: глутамат, АТФ, Д-серин.

В данном приближении мы не описываем промежуточные процессы генерации кальциевых импульсов, зависящие от собственной динамики астроцита. В стохастической версии модели концентрация глутамата описывается случайным пуассоновским процессом, а концентрация глиатирансмиттера линейно влияет на частоту генерации данного пуассоновского процесса.

Для простейшей реализации данной модели с одним глиатрансмиттером – глутаматом, который может изменять вероятность пресинаптического выброса, удалось получить ряд интересных динамических режимов. Известно, что в зависимости от рецепторной композиции пресинаптической терминали, астроцитарный глутамат может как увеличивать, так и уменьшать вероятность выброса. Эти эффекты, соответственно, приводят к астроцитарной потенциации и депрессии синаптической передачи.

В случае астроцитарной потенциации для простой стохастической модели был обнаружен эффект бистабильности за счет положительной обратной связи нейротрансмиттер-астроцит. Зависимость вероятности выброса нейротрансмиттера в таком режиме показана на рис. 1.

В случае астроцитарной депресии, когда вероятность выброса уменьшается в присутствии глиального глутамата, наблюдалось формирование транзиентных импульсов в зависимости вероятности выброса от времени, как показано на рис. 2.

В усредненной модели нейрон-глиального взаимодействия динамику системы можно представить на фазовой плоскости, как это показано на рис. 3.

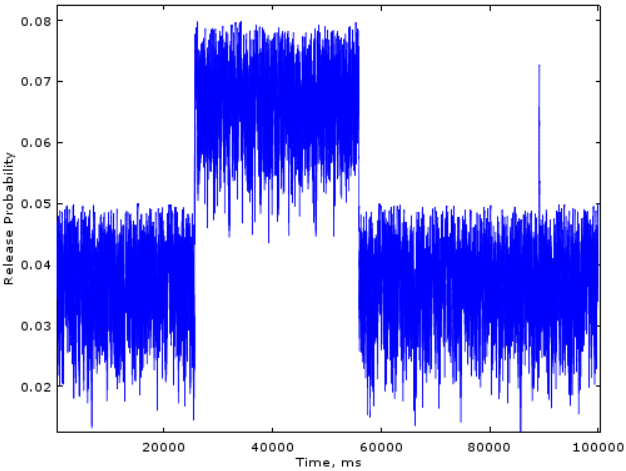


Рис. 1. Бистабильная динамика в стохастической модели одиночного трехчастного синапса

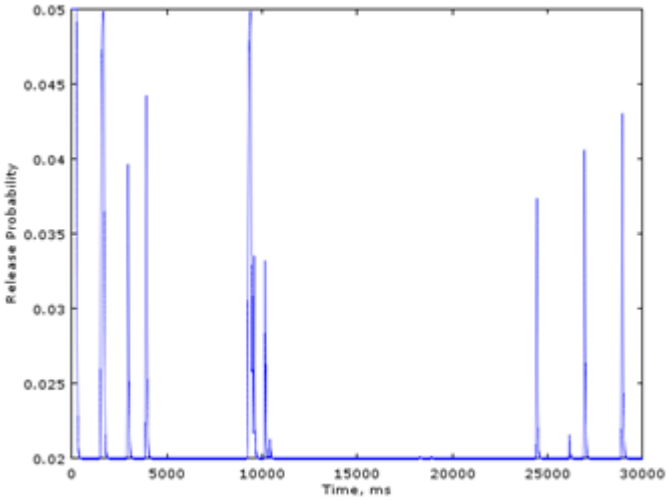


Рис. 2. Импульсная динамика вероятности выброса нейротрансмиттера от времени при астроцитарной депрессии и отсутствии эффектов кратковременной пластичности синапса

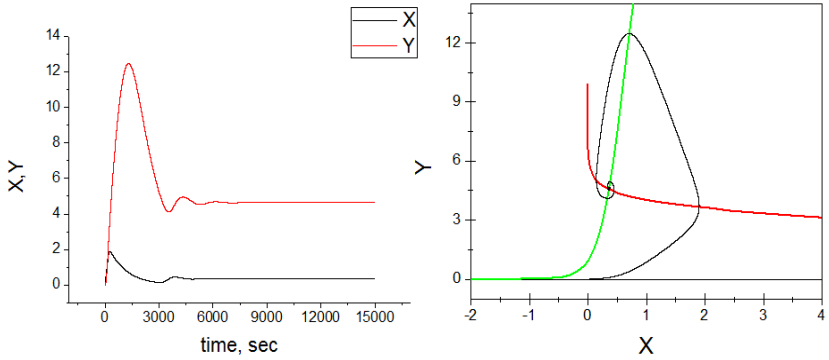


Рис. 3. Качественный анализ динамики в системе нейрон-глиального взаимодействия при астроцитарной депрессии. Временные зависимости (слева) и динамика на фазовой плоскости (справа)

Модель нейрон-глиальной сети

Рассмотрим теперь нейрон-глиальную сеть с трехчастными синаптическими контактами, рассмотренными в предыдущем разделе. В качестве модели одиночного нейрона сети будет использовать модель Ижикевича [8]. Модельные уравнения будут иметь следующий вид:

$$\frac{dV}{dt} = 0.04V^2 + 5V + 140 - u + I,$$
$$\frac{du}{dt} = a(bv - u), \qquad (3)$$
$$\text{если } V \geq V_{peak}, \begin{cases} V \leftarrow c, \\ u \leftarrow u + d. \end{cases}$$

Здесь V – потенциал мембраны нейрональной клетки, измеряемый в мВ, t – время в мс, u – «переменная восстановления», описывающая процесс активации и деактивации калиевых и натриевых мембранных каналов соответственно. Параметр a устанавливает характерный временной масштаб изменения переменной восстановления u. Значение V_{peak} ограничивает амплитуду спайка. Параметры c и d задают значения V и u после генерации спайка. I – синаптический ток. Веса синаптических связей между нейронами задаются матрицей S = (s_{ij}), где генерация спайка j-м нейроном мгновенно изменяет значение потенциала V на величину s_{ij}.

В связи с тем, что регуляция нейронной активности астроцитами осуществляется на сравнительно больших временных масштабах (порядка секунд), нежели генерация спайков, был введен средний уровень активности,

Q, характеризующий динамику отдельных нейронов и описывающийся следующими уравнениями:

$$\frac{dQ}{dt} = -\alpha_q Q + \beta_q H_q(V), \quad H_q(V) = \frac{1}{1+\exp(-V/k_q)}, \tag{4}$$

где α_q – константа времени релаксации, β_q коэффициент шкалирования с $0 < \alpha_q << \beta_q$, k_q – наклон активационной функции $H_q(V)$, $k_q << 1$.

Каждый сгенерированный спайк на нейроне будет сопровождаться высвобождением нейротрансмиттера глутамата. С целью упрощения вычислений будет использоваться феноменологический подход для описания динамики концентрации высвобожденного нейротрансмиттера. Для этих целей предлагается использование метода среднего поля, позволяющего описать изменение концентрации нейротрансмиттера в следующей форме:

$$\frac{dX}{dt} = -\alpha_x X + k_{pre}\beta_x H_x, \qquad H_x(Q) = \frac{1}{1+\exp(-(Q-\theta_x)/k_x)}, \tag{5}$$

где α_x – константа релаксации, $\beta_x H_x(Q)$ – доля высвобожденного нейротрансмиттера при генерации спайка на нейроне, $H_x(Q)$ – функция активации, k_{pre} – эффективность высвобождения нейротрансмиттера.

С целью упрощения вычислений будет использоваться метод среднего поля, позволяющий описать динамику концентрации глиатрансмиттера в следующей форме:

$$\frac{dY_k}{dt} = -a_k Y_k + \beta_k H_k(X_e),$$
$$H_k(X_e) = \frac{1}{1+\exp(-\frac{(X-\theta_k)}{k_k})}, \tag{6}$$

где k = глутамат, D-серин, Y_k – концентрация глиатрансмиттера, $H_k(X_e)$ – функция активации.

Воздействие астроцита будет результироваться в изменении синаптических весов связей между нейронами следующим образом:

$$k_s = 1 + \gamma_s Y_k,$$
$$S = k_s \sum s_{ij}, \tag{7}$$

где k_s – параметр, определяющий изменение синаптических весов нейрона под воздействием глиатрансмиттера, γ_s – параметр, определяющий воздействие астроцита на нейронную активность.

Схематически модель представлена на рис. 4. В расчетах использовалась сеть размерностью 10000 нейронов с 1 000 000 синаптических связей.

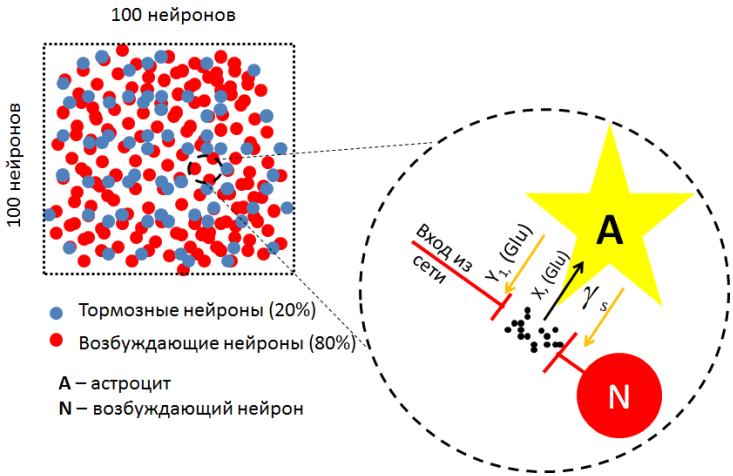


Рис. 4. Схема математической модели регуляции нейронной активности в нейрон-глиальной сети. Астроцит может оказывать воздействие на нейронную активность через высвобождение глутамата, Y_1, и D-серина, Y_2.

Основываясь на анатомии коры головного мозга, соотношение возбуждающих нейронов к тормозным было выбрано 4 к 1. В модели тормозные синаптические связи являются сильными. Помимо синаптических входов, каждый нейрон стимулируется гауссовским шумом.

В результате вычислительных экспериментов было установлено, что активация астроцитов приводит к появлению квазисинхронных пачечных разрядов (рис. 5), что соответствует экспериментальным гипотезам и существующим теоретическим работам [8].

Выводы

Таким образом, с использованием модели трехчастного синапса мы показали, что обратная связь, формируемая между нейронами и астроцитами, в нейрон-глиальной сети может существенно влиять на эффективность передачи импульсов между нейронами. В частности, формирование в трехчастном синапсе положительной обратной связи приводит к возникновению синаптической бистабильности – сосуществованию двух стационарных уровней нейрональной активности. Этот эффект достаточно хорошо качественно согласуется с формированием персистентной активности в префронтальной коре головного мозга [6].

В случае отрицательной обратной связи в синапсе происходит возникновение всплесков вероятности проведения импульсов. Интересно, что на сетевом уровне такие всплески приводят к возникновению квазисинхронных пачечных разрядов. Параметры пачечных разрядов (частота и длительность)

в таком случае будут зависеть от величины изменений кратковременной пластичности. Этот эффект хорошо согласуется с электрическими разрядами в нейрон-глиальных культурах in vitro [9].

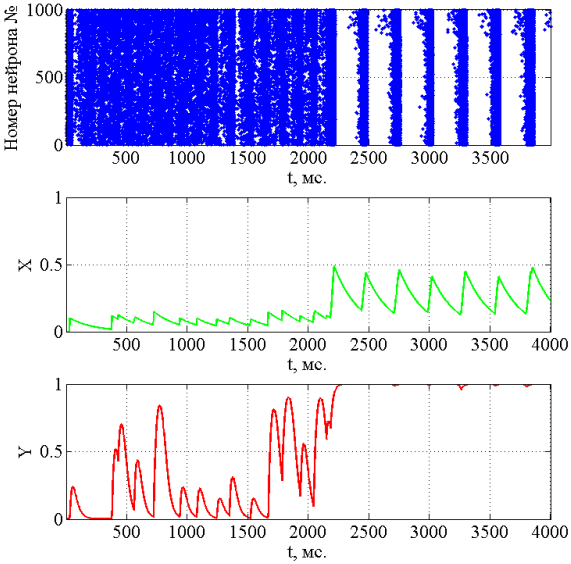


Рис. 5. Формирование квазисинхронных пачечных разрядов в нейрон-глиальной сети. Сверху-вниз: зависимость нейронной активности всей сети, концентрации нейротрансмиттера (X) и глиатрансмиттера (Y) для синаптических связей нейрона № 10 от времени

Следует отметить, что наблюдаемые эффекты (возникновение бистабильного и импульсного режимов в модели) не связаны со сложностью локальной динамики нейронов и астроцитов, а определяются существованием петли обратной связи между пресинаптическим нейроном и астроцитом. Также было показано, что активация астроцитов соответствует переходу нейрон-глиальной сети в режим квазисинхронной активности.

Работа подержана Министерством науки и высшего образования РФ, проект № 14.Y26.31.0022.

Список литературы

1. Araque A. [et al.]. Glutamate-dependent astrocyte modulation of synaptic transmission between cultured hippocampal neurons // Eur. J. Neurosci. 1998. V. 10, N 6. P. 2129–2142.
2. Araque A. [et al.]. Tripartite synapses: glia, the unacknowledged partner // Trends

Neurosci. 1999. V. 22, N 5.
3. Wittenberg G.M., Sullivan M.R., Tsien J.Z. Synaptic reentry reinforcement based network model for long-term memory consolidation // Hippocampus. 2002. V. 12, N 5. P. 637–647.
4. Wang X.J. Synaptic basis of cortical persistent activity: the importance of NMDA receptors to working memory // J. Neurosci. 1999. Vol. 19, № 21. P. 9587–9603.
5. Wang X.-J. Neural dynamics and circuit mechanisms of decision-making // Curr. Opin. Neurobiol. Elsevier Current Trends, 2012. Vol. 22, № 6. P. 1039–1046.
6. Mongillo G., Barak O., Tsodyks M. Synaptic theory of working memory // Science. 2008. V. 319, N 5869. P. 1543–1546.
7. Compte, A. [et al.]. Temporally Irregular Mnemonic Persistent Activity in Prefrontal Neurons of Monkeys During a Delayed Response Task // J. Neurophysiol. 2003. V. 90, N 5. P. 3441–3454.
8. Pereira A., Furlan F.A. On the role of synchrony for neuron-astrocyte interactions and perceptual conscious processing // J. Biol. Phys. 2009. Vol. 35, N 4. P. 465–480.
9. Pimashkin A. [et al.]. Spiking Signatures of Spontaneous Activity Bursts in Hippocampal Cultures // Front. Comput. Neurosci. 2011. V. 5, N November. P. 1–12.

КОГНИТИВНЫЕ НАУКИ И ИНТЕРФЕЙС «МОЗГ-КОМПЬЮТЕР»

С. А. ПОЛЕВАЯ[1,3], Н. А. БУЛАНОВ[2], С. Б. ПАРИН[3]
[1] Приволжский исследовательский медицинский университет, Нижний Новгород
[2] Национальный исследовательский университет
«Высшая школа экономики», Москва
[3] Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского
p453383@mail.ru

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ ДЛЯ СКРИНИНГА, ДИАГНОСТИКИ И ЦИФРОВОГО ОТОБРАЖЕНИЯ КОГНИТИВНЫХ НАРУШЕНИЙ *

В рамках концепции цифровой психофизиологии рассматриваются современные интернет-ориентированные компьютерные технологии, обеспечивающие объективный мониторинг текущего состояния когнитивных функций, а также скрининг, диагностику и коррекцию когнитивных нарушений. Дана краткая характеристика функциональных возможностей Web-платформы ApWay.ru, разработанной с участием авторов для этих целей, и приводятся примеры ее апробации и использования.

Ключевые слова: когнитивные функции, мониторинг, скрининг, диагностика, реабилитация, Web-платформа ApWay.ru.

Введение

В современной неврологии и нейропсихологии широко используются методы диагностики и коррекции когнитивных функций, основанные на традиционных подходах. Эти методы предполагают присутствие активного эмоционального и мотивационного компонента как у специалистов- неврологов, так и у их пациентов. При наличии этих условий пациент вовлекается в выполнение диагностических задач и реабилитационных процедур. Однако такой подход имеет целый ряд существенных недостатков. Во-первых, он полностью зависит от профессиональной квалификации, когнитивных и аффективных ресурсов и текущего функционального статуса человека-эксперта и, соответственно, подвержен искажениям сугубо субъективного свойства. Во-вторых, при этом традиционном подходе чрезвычайно ограничено простран-

* Данная работа выполнена при частичной поддержке РФФИ, проекты № 18-013-01225_а, 18-413-520006_р_а, 19-013-00095_а.

ство признаков, описывающих структуру индивидуальной когнитивной системы пациента. В-третьих, для данного подхода характерна чрезвычайно низкая точность детектирования признаков нарушений в когнитивной сфере. Таким образом, очевидно, что в рамках принятых клинических стратегий отсутствует возможность для объективного цифрового картирования и контролируемой оптимизации когнитивных функций. Это существенно снижает эффективность диагностики когнитивных нарушений и реабилитации пациентов с такими нарушениями.

Современное состояние вопроса

В последние годы появились возможности для значительного прорыва в обсуждаемой области. Компьютерные технологии, включая технологии виртуальной реальности и соответствующий программный инструментарий, позволяют объективизировать диагностику когнитивных нарушений и сделать более прицельными способы их коррекции [1]. В основе этих технологий лежат базовые принципы физических измерений, заключающиеся в сравнении объекта с эталоном, в качестве которого при измерениях свойств субъективного когнитивного пространства человека можно рассматривать информационные объекты и событийные контексты виртуальной компьютерной среды.

Таким образом, процедура измерения сводится к формализованным оценкам ошибок распознавания, управления или воспроизведения виртуальных эталонов. Результатом этих измерений становятся цифровые когнитивные карты, дающие объективное отображение когнитивной системы конкретного человека в широком диапазоне параметров когнитивного процесса [2, 3].

На сегодняшний день созданы локальные и интернет-ориентированные программные инструменты для когнитивной диагностики и реабилитации (Wikium.ru, Lumosity.com, Cognifit.com, Platform.apway.ru и др.). Они обеспечивают объективные измерения и различные виды тренировки внимания, памяти, восприятия, быстродействия, гибкости по отношению к виртуальным объектам в различных событийных контекстах [4–5]. Как правило, тестирование проводится по отношению к визуальным объектам разной семантики (геометрические фигуры, буквы, слова, предметные изображения, лица с разнообразной эмоциональной экспрессией) в ограниченном наборе событийных контекстов. Особенность этих программных инструментов заключается в том, что событийные контексты обеспечивают выполнение заданий по узнаванию, поиску и сравнению визуальных образов [7–9]. При этом происходит измерение разных типов ошибок распознавания, времени реакции, психофизических порогов обнаружения и различения (дифференциальных и нижних абсолютных порогов). Базовыми моделями для реализации тестов являются тест Струпа, методика Вундта, тест «Фигуры Готтшальдта » и подобные им психофизические методики в разных модификациях.

Как правило, в современных компьютерных диагностико-реабилитационных тренажерах реализация тестов осуществляется в форме увлекательных компьютерных игр. При этом обязательным элементом всех тренажеров является обратная связь с цифровой оценкой тренируемых функций и отображением истории тренировок в форме временно́й диаграммы оценок. Спортивный и игровой азарт мотивирует пользователя к продолжительным занятиям на когнитивных тренажерах, что увеличивает ресурсы для сбора диагностических данных и реабилитационных мероприятий. Однако при каждой игре актуализируется множество когнитивных процессов, что, напротив, затрудняет дифференциальную диагностику и коррекцию поврежденного когнитивного модуля.

Альтернативой когнитивным тренажерам, массово представленным в открытом Интернет-пространстве, служит экспертная система, расположенная на web-платформе ApWay (platform.apway.ru), разработанная в Приволжском исследовательском медицинском университете (Н. Новгород), Нижегородском государственном университете им. Н.И. Лобачевского и Высшей школе экономики (Москва). Она обеспечивает возможности для цифрового картирования когнитивных функций в широком пространстве признаков и предоставляет удобный интерфейс для конструирования оригинальных пользовательских тестов [10]. К настоящему времени на платформе размещено более 350 сценариев, позволяющих проводить измерения отдельных когнитивных модулей в трех целевых контекстах: сенсомоторная активность по широкому набору визуальных признаков и объектов; поиск объекта; ассоциации разномодальных информационных образов.

Так, в среде ApWay, на базе оригинальной авторской модели теста «компьютерная кампиметрия» [11], создана уникальная инфраструктура для реализации тестирования функции выделения визуальных объектов из фона. В сценариях тестов, построенных по этой модели, могут использоваться визуальные объекты с различной семантикой (буквы, слова, предметные изображения, геометрические фигуры), локализованные в разных зонах экрана. Перед пользователем сначала ставится задача проявить фигуру на цветовом фоне, указать на пиктограмму этой фигуры, а затем снова спрятать фигуру. Одна и та же последовательность событий для разных оттенков фона позволяет построить психофизическую функцию цветоразличения, которая является цифровой картой субъективного цветового пространства и отображает особенности цветоразличения данного конкретного пользователя. Данная методика располагает разнообразным функционалом и, например, открывает возможность для инструментальной диагностики зрительной объектной агнозии независимо от речевых и мнестических функций.

В основу теста сенсомоторной активности положен классический метод измерения сенсомоторной реакции. Однако анализ связи между сенсорными

и моторными событиями принципиально реализован в рамках парадигмы активности, допускающей действия, связанные не только с прошедшими, но и с предсказанными будущими событиями. Возможности модуля позволяют задавать различные параметры испытания, такие как: выбирать вид стимула (векторные изображения, картинки, текст); определять целевые стимулы, место расположения стимулов на экране, время экспозиции для каждого стимула, межстимульный интервал, фон рабочей области, задержку перед началом испытания. Таким образом, «конструктор» платформы ApWay.ru позволяет создавать большое разнообразие сценариев для изучения простой и сложной сенсомоторной активности. В совокупности показатели данного теста позволяют определить степень сохранности отделов головного мозга, особенности распознавания цвета, уровень сенсомоторной интеграции, ресурсы пространственного и селективного внимания, способность к обучению и прогнозированию.

Классический тест Струпа позволяет спровоцировать ситуацию когнитивного конфликта. Отличительной особенностью теста, реализованного на WEB-платформе ApWay.ru, является возможность задавать названия цветов на различных языках или добавлять собственные варианты цвета для проведения экспериментов. По результатам измерений фиксируются ошибки и время реакции при решении каждой задачи, после первичного анализа данных система выдаёт информацию о количестве ошибок в каждом контексте и среднем времени решения задачи в каждом контексте. На основании этих показателей делается вывод об уровне когнитивного контроля, эффективности обработки вербальной и цветовой информации, а также об уровне стрессоустойчивости при ментальном стрессе.

Отличительной особенностью когнитивной платформы ApWay.ru является то, что каждый тест на ней может быть оптимизирован для актуализации отдельных когнитивных модулей и обеспечивать как построение персональных цифровых когнитивных карт, так и формирование на их основе индивидуальных программ когнитивных тренировок. Необходимо признать, что существенным недостатком этой технологии на настоящем уровне реализации является недостаточно удобный пользовательский интерфейс для обратной связи и дистанционного мониторинга процесса реабилитации.

В целом на сегодняшний день созданы технологические предпосылки для объективизации когнитивной диагностики и реабилитации. Разрабатываются шаблонные цифровые карты по базовым признакам информационных объектов (пространственные, временные, количественные, качественные) и базовым когнитивным процессам (выделение признаков, идентификация и классификация объектов, селективное внимание, принятие решений) для основных нейропсихологических нарушений. Уникальные возможности по

персонализированной цифровизации когнитивной системы пока слабо реализованы. Однако следует признать, что происходит активное движение в этом направлении.

Результаты

В настоящее время интернет-платформы, предоставляющие когнитивные тренажеры, успешно опробованы в клинике, например, для диагностики и реабилитации пациентов с деменцией, для постинсультной реабилитации больных с очаговыми нарушениями мозга [12], а также для восстановления когнитивных функций у онкологических больных после прохождения химиотерапии [13]. Кроме того, они многократно демонстрировали широкие возможности для мониторинга состояния когнитивных функций при решении сложных, а иногда и экстремальных когнитивных задач. Интернет-платформа ApWay.ru была успешно апробирована при мониторинге динамики когнитивных функций у школьников и студентов в процессе обучения, у детей с синдромом дефицита внимания и гиперактивности, у сотрудников силовых структур и в других контекстах. В качестве примера приведем результаты экспериментального исследования особенностей когнитивной и вегетативной сферы у переводчиков синхронистов, выполненного нами под руководством профессора Т.В. Черниговской в рамках гранта РФФИ 16-06-00501_а в 2016–2018 годах.

В экспериментах приняли участие 33 испытуемых. Экспериментальную группу («профи») составили 22 лингвиста, прошедших специальное обучение синхронному переводу и имеющих опыт в этой сфере деятельности (17 женщин и 5 мужчин, возраст – от 20 до 34 лет). В контрольной серии (n = 11, 8 женщин и 3 мужчин в том же возрастном диапазоне) участвовали лица, владеющие иностранным языком, но не имеющие навыков синхронного перевода. В процессе эксперимента все испытуемые выполняли сложные профессиональные задачи по синхронному переводу докладов участников Всемирного экологического форума с иностранного (английского и немецкого) языка на русский и обратно (с русского на иностранный), а также выполняли тестовые задания по повтору текстов. До и после выполнения профессиональных задач переводчики проходили психофизиологические тесты, размещенные на платформе ApWay: компьютерную кампиметрию, тест на простую сенсомоторную активность и тест Струпа. Для субъективной оценки функционального состояния использовали проективный тест на уровень эмоциональной дезадаптации (УЭД) [14].

Проведенное до и после выполнения лингвистических заданий психофизиологическое тестирование позволило выявить ряд особенностей динамики когнитивно-аффективной сферы при синхронном переводе. Так, по результатам проективного теста УЭД в экспериментальной группе было зарегистрировано незначительное, но статистически значимое ($p < 0,05$) увеличение уровня эмоциональной дезадаптации после решения лингвистических

задач, тогда как в контрольной группе эти изменения были статистически не достоверны. В обеих группах по тесту компьютерной кампиметрии не удалось обнаружить существенных изменений в уровне дифференциальных порогов цветоразличения, что позволяет говорить об относительной стабильности базового функционального состояния. При выполнении простого сенсомоторного теста синхронные переводчики как до, так и, особенно, после решения лингвистических задач проявили статистически значимо меньшую скорость реакции, но зато существенно большую ($p < 0,01$) точность, что позволяет говорить о высоком уровне когнитивного контроля у них по сравнению с контрольной группой.

Данная ситуация подтвердилась и в тесте Струпа (таблица 1). Этот компьютеризированный тест, имитирующий когнитивный конфликт, на наш взгляд, является адекватной экспериментальной моделью сложных когнитивных нагрузок, позволяющей воспроизводить различные варианты взаимодействия информационных образов: как их консолидацию (физиологический аналог – облегчение в нейронных модулях), так и конкуренцию (в нейронных сетях – окклюзия).

Таблица 1

Средняя по группам продолжительность решения задач теста Струпа до и после выполнения лингвистических заданий (с, M±m)

ЭТАП	ГРУППА	mono	color	True Text	True Color
до	«профи»	1016,96 ±248,70	930,62 ±185,63*	1175,48 ±373,01*	1158,04 ±410,01*
	контроль	1264,68 ±534,20	1071,35 ±251,29*	1369,26 ±464,02*	1264,68 ±534,20
после	«профи»	985,87 ±241,22	921,50 ±191,15*	1149,45 ±390,36*	1127,19 ±471,37*
	контроль	1356,71 ±894,80	1147,83 ±714,67*	1453,95 ±845,00	1486,52 ±889,63*

* - достоверные ($p < 0,05$) различия показателей с задачей «mono»

Характерно, что как в группе профессиональных переводчиков, так и в группе «контроль» до и после выполнения лингвистических заданий проявляется выраженный эффект консолидации: при идентичности вербального и сенсорного образов цвета (задача color, цвет букв соответствует смыслу слова) уменьшается время принятия решения относительно задачи с черно-белыми названиями цветов (задача mono).

Межгрупповые отличия во взаимодействии разномодальных образов цвета обнаруживаются в задачах с когнитивным конфликтом: у профессио-

нальных переводчиков до и после лингвистических заданий проявляется эффект конкуренции: при рассогласовании вербальных и сенсорных образов (цвет букв слова не соответствует смыслу слова) увеличивается время принятия решения относительно задачи с черно-белыми названиями цвета (задача mono), то есть затрудняется выбор решения как по смыслу слова (задача True text), так и по цвету букв (задача True color); в группе «контроль» до и после лингвистических задач проявляется инверсия эффектов конкуренции: до лингвистических заданий затрудняется выбор по смыслу слова (задача true text), то есть сенсорный образ тормозит реализацию вербального образа; после же лингвистических заданий затрудняется выбор по цвету букв, то есть вербальный образ тормозит реализацию сенсорного образа. Эти результаты позволяют высказать предположение, что в контрольной группе после выполнения лингвистических задач происходит усиление вербальных образов, на фоне ослабления активности сенсорных образов.

Заключение

Таким образом, разработанные в последние годы Интернет-ориентированные компьютерные технологии позволяют не только регистрировать объективные показатели текущего функционального состояния когнитивных функций, но и выявлять устойчивые маркеры их нарушений и формировать эффективные программы когнитивной реабилитации. Дополнительные возможности в этом направлении предоставляет Интернет-ориентированная технология событийно-связанной телеметрии ритма сердца, позволяющая в реальном режиме времени регистрировать и анализировать показатели вегетативного обеспечения когнитивных процессов и оценивать взаимосвязь этих показателей с уровнем когнитивной сложности когнитивных задач [15]. В перспективе новые принципы формализованного описания индивидуальных когнитивных систем могут привести к пересмотру сложившейся классификации когнитивных нарушений и созданию принципиально новых моделей когнитивной диагностики и реабилитации [16].

Список литературы

1. Величковский Б.М., Соловьев В.Д. Компьютеры, мозг, познание: успехи когнитивных наук. Москва : Наука; 2008. 293 с.
2. Morrison G.E., Simone C.M., Ng N.F., Hardy J.L. Reliability and validity of the NeuroCognitive Performance Test, a web-based neuropsychological assessment // Front. Psychol. 2015. V. 6. P. 1652.
3. Jiang T. Brainnetome: a new-ome to understand the brain and its disorders // Neuroimage. 2013. V. 80. P. 263–272.
4. Feenstra H.E.M., Murre J.M.J., Vermeulen I.E., Kieffer J.M., Schagen S.B. Reliability and validity of a selfadministered tool for online neuropsychological testing: the Amsterdam Cognition Scan // J. Clin. Exp. Neuropsychol. 2018. V. 40, N 4. P. 253–273.

5. Fliessbach K., Hoppe C., Schlegel U., Elger C.E., Helmstaedter C. NeuroCogFX — a computer-based neuropsychological assessment battery for the follow-up examination of neurological patients // Fortschr. Neurol. Psychiatr. 2006. V. 74, N 11. P. 643–650.
6. Guimarães B., Ribeiro J., Cruz B., Ferreira A., Alves H., Cruz-Correia R., Madeira M.D., Ferreira M.A. Performance equivalency between computer-based and traditional pen-andpaper assessment: a case study in clinical anatomy. // Anat. Sci. Educ. 2018. V. 11. N 2. P. 124–136.
7. Tsotsos L.E., Roggeveen A.B., Sekuler A.B., Vrkljan B.H., Bennett P.J. The effects of practice in a useful field of view task on driving performance //Journal of Vision. 2010. V. 10, N 7. P. 152–152.
8. Crabb D.P., Fitzke F.W., Hitchings R.A., Viswanathan A.C. A practical approach to measuring the visual field component of fitness to drive // Br. J. Ophthalmol. 2004. V. 88, N 9. P. 1191–1196.
9. Edwards J.D., Vance D.E., Wadley V.G., Cissell G.M., Roenker D.L., Ball K.K. Reliability and validity of useful field of view test scores as administered by personal computer // J. Clin. Exp. Neuropsychol. 2005. V. 27, N 5. P. 529–543.
10. Полевая С.А., Мансурова (Ячмонина) Ю.О., Ветюгов В.В., Федотчев А.И., Парин С.Б. Особенности когнитивных функций и их вегетативного обеспечения при нарушениях эндогенной опиоидной системы // XX Международная научно-техническая конференция «Нейроинформатика–2018». М: НИЯУ МИФИ. 2018. Ч. 2. С. 162–170.
11. Полевая С.А., Парин С.Б., Стромкова Е.Г. Психофизическое картирование функциональных состояний человека. // Экспериментальная психология в России: традиции и перспективы / под ред. Барабанщикова В.А. Москва : Изд-во «Институт психологии РАН». 2010. С. 534–538.
12. Shatil E., Mikulecká J., Bellotti F., Bureš V. Novel television-based cognitive training improves working memory and executive function // PLoS One. 2014. V. 9, N 7. e101472,
13. Bray V.J., Dhillon H.M., Bell M.L., Kabourakis M., Fiero M.H., Yip D., Boyle F., Price M.A., Vardy J.L. Evaluation of a web-based cognitive rehabilitation program in cancer survivors reporting cognitive symptoms after chemotherapy // J. Clin. Oncol. 2017. V. 35, N 2. P. 217–225.
14. Григорьева В.Н., Тхостов А.Ш. Способ оценки эмоционального состояния человека / Патент РФ 2291720. 2007.
15. Polevaya S.A., Eremin E.V., Bulanov N.A., Bakhchina A.V., Kovalchuk A.V., Parin S.B. Event-related telemetry of heart rhythm for personalized remote monitoring of cognitive functions and stress under conditions of everyday activity // Sovremennye tehnologii v medicine. 2019. V. 11, N 1. P. 109–115,
16. Полевая С.А., Рунова Е.В., Некрасова М.М., Федотова И.В., Бахчина А.В., Ковальчук А.В., Шишалов И.С., Парин С.Б. Телеметрические и информационные технологии в диагностике функционального состояния спортсменов // Современные технологии в медицине. 2012. Т. 4, № 4. С. 94–98.

С. А. ТЕРЕХИН, К. В. СИДОРОВ
Тверской государственный технический университет
rabeenovich69@mail.ru, bmisidorov@mail.ru

ПОКАЗАТЕЛИ АТТРАКТОРОВ БИОМЕДИЦИНСКИХ СИГНАЛОВ, ХАРАКТЕРИЗУЮЩИХ ЭМОЦИОНАЛЬНЫЕ РЕАКЦИИ ЧЕЛОВЕКА*

Исследована возможность анализа динамики эмоциональной реакции человека на основе невербальной информации. Описаны методика и биотехническая система для мониторинга эмоциональных реакций. По полученным паттернам ЭЭГ реконструированы аттракторы. Проведена оценка показателей аттракторов на различных этапах эмоциональной стимуляции. Проанализировано влияние фактора зашумленности исходных данных. По результатам анализа ЭЭГ отмечено сохранение общей тенденции в развитии динамики эмоций.

Ключевые слова: эмоция, эмоциональная реакция, аттрактор, ЭЭГ, ЭМГ, биотехническая система.

Введение

Эмоциональные реакции человека связаны с деятельностью мозга и проявляются в особенностях работы его отдельных функциональных подсистем. В процессе изучения механизма формирования эмоциональных феноменов неоднократно предпринимались попытки его формализации, а также построения различных моделей эмоций [1, 2].

В последние полтора-два десятилетия с развитием аппаратно-программных комплексов для мониторинга и контроля функционального состояния человека возрастают возможности количественных оценок отдельных характеристик эмоций испытуемого по разнотипным биомедицинским сигналам (частота сердечных сокращений (ЧСС), кожно-гальваническая реакция (КГР), электромиограмма (ЭМГ), электрокардиограмма (ЭКГ), электроэнцефалограмма (ЭЭГ), образцы речевых сигналов и др.). Модели эмоций, основанные на анализе подобных биомедицинских сигналов, применяются как в исследовательских целях, так и для моделирования эмоциональных реакций в искусственных средах (при анимации, в компьютерных играх и др.). Среди математических моделей можно особо выделить следующие формализованные описания: ACRES/WILL, OCC, KARO, EMA, MAMID, Soar-Emote,

* Данная работа выполнена при финансовой поддержке РФФИ в рамках научного проекта № 18-37-00225.

TABASCO, WASABI, Affective Computing, модель Фоминых И.Б.–Леонтьева В.О., модель Глазунова Ю.Т. [3, 4].

В общем виде описания эмоций в вышеприведенных моделях можно привести к выражению: $M = <X_1, X_2, \ldots, X_n>$, где X_n – n-я переменная (характеристика), описывающая некоторое свойство эмоциональной реакции. Построение моделей предоставит возможности формирования пространства признаков для интерпретации эмоциональных реакций. Эта задача может быть решена путем использования взаимосвязи некоторых биомедицинских сигналов и требуемых характеристик эмоций.

В настоящее время существуют потребности в создании биотехнических систем, ориентированных на задачи мониторинга отдельных характеристик эмоциональных реакций испытуемых. Важным моментом в данных исследованиях является подбор математического аппарата, который позволит эффективно анализировать зарегистрированные биомедицинские сигналы, учитывая фактор зашумленности исходных данных.

Методика исследования эмоциональных реакций человека

В работах [5, 6] описывается биотехническая система «EEG/S» для исследования эмоциональных реакций. Предложена модель интерпретатора эмоций человека (ME) на основе мониторинга ЭЭГ-сигналов и паттернов речевых сигналов, которая включает три компонента:

$$ME = <Z, U, D>, \quad (1)$$

где Z – знак эмоциональной реакции (положительный, отрицательный, нейтральный); U – уровень эмоциональной реакции (низкий, средний, высокий); D – динамика эмоциональной реакции (тренд возрастает, тренд убывает, тренд без изменений).

Алгоритмы оценки компонентов Z и U (1) подробно обсуждались в работе [4]. Для формирования третьего компонента модели D (1) в методику работы с биотехнической системой «EEG/Speech» были внесены изменения, которые позволяют интерпретировать характер изменения свойств эмоциональной реакции на протяжении регистрируемого интервала времени:

- запись речевых сигналов должна осуществляться непрерывно и параллельно с регистрацией ЭЭГ и ЭМГ-сигналов;
- записи сигналов обязаны содержать полные протоколы экспериментов (реакции испытуемых, комментарии с самооценками стимулов, звуковые и видеотреки стимулов (для зрительных или слуховых каналов), а также метки начала и конца стимулов;
- определение канала для предъявления внешних стимулов должно осуществляться по результатам предварительной оценки чувствительности анализаторов у испытуемого;
- регистрация реакций испытуемых обязана проводиться с помощью контрольных фраз, содержащих основные гласные фонемы;

- каждый испытуемый должен ознакомиться с образцами стимулов, и для него должен быть сформирован и настроен индивидуальный сценарий исследования.

Таким образом, в своих исследованиях мы используем многоканальную биотехническую систему «EEG/Speech+». В данном типе экспериментов были задействованы четыре канала регистрации сигналов: канал ЭЭГ – электроэнцефалограф «Энцефалан-131-03» (ООО НПКФ «Медиком МТД», г. Таганрог), канал ЭМГ – миограф «Нейро-МВП-4» (ООО «Нейрософт», г. Иваново), канал речи и информационный канал отчета.

Регистрация ЭЭГ проводилась в соответствии с международной системой «10-20» [7] по 19 отведениям, частота дискретизации составляла 250 Гц. Сигнал ЭМГ снимался с левой стороны лица в каналах «corrugator supercilii» и «zygomaticus major» в соответствии с методикой «Fridlund and Cacioppo» [8]. Запись сигнала ЭМГ велась при частоте дискретизации 1 000 Гц. Регистрация речевых сигналов велась при частоте дискретизации 22 050 Гц и разрешении 16 бит.

Участниками экспериментов стали 10 мужчин в возрасте 20–30 лет. Каждый испытуемый дал согласие на участие в эксперименте и был проинструктирован о ходе задач и своих обязанностях. Эксперименты проходили в тихом помещении, в дневное время суток, испытуемый располагался в удобном кресле. К каждому испытуемому применена аудиовизуальная стимуляция, все стимулы разбиты на три группы эмоциональных реакций: положительные эмоции (удовольствие) – Э(+); отрицательные эмоции (грусть, отвращение, страх) – Э(–); нейтральное состояние – Э(Н).

Для дальнейшего анализа все зарегистрированные биомедицинские сигналы сегментируются на безартефактные паттерны длительностью 8 секунд.

Математический аппарат для анализа биомедицинских сигналов

В ходе анализа биомедицинских сигналов исследователи сталкиваются с большим объемом поступающей информации. Возникает необходимость в систематизации получаемых данных и выделении характерных информативных признаков. Для решения данной задачи в настоящее время активно применяются методы нелинейной динамики, предполагающие реконструкцию аттракторов на основе отдельных паттернов биомедицинских сигналов [9-12].

В основе вышеприведенного подхода лежит теорема Ф. Такенса, согласно которой можно восстановить фазовый портрет динамической системы по скалярному временному ряду [13]. Для реконструкции аттрактора используется метод «Grassberger and Procaccia» [14], который позволяет формировать векторы в новом фазовом пространстве на основе значений временного ряда, взятых с запаздыванием:

$$y_n = (x_n, x_{n+\tau}, x_{n+(m-1)\tau}),\ n = 0,...,s-1,\ s = N-(m-1)\tau\tau, \qquad (2)$$

где τ – временной лаг; m – размерность вложения; N – общее число точек временного ряда.

Результаты реконструкции зависят от численных значений параметров τ и m (2). Очевидно, что значения τ и m связаны с характером изменения временного ряда. Поэтому для нахождения временного лага τ часто используют метод минимальной корреляции между отдельными значениями временного ряда. Для этого вычисляется автокорреляционная функция и выбирается лаг, соответствующий времени пересечения нуля.

Для нахождения оптимального значения параметра m существует несколько способов, среди которых можно выделить метод, основанный на вычислении корреляционного интеграла [15, 16]:

$$C_m(r) = \lim_{N\to\infty} \frac{1}{N^2} \sum_{k,l=1}^{N} \theta(r - dd(k,l)), \qquad (3)$$

где N – количество точек аттрактора; θ – ступенчатая функция Хэвисайда; r – длина грани m-мерного куба; $dd(k,l)$ – расстояние между k-й и l-й точками аттрактора.

На основе корреляционного интеграла (3) определяется размерность D_2:

$$D_2 = \lim_{r\to 0} \log C_m(r) / \log r. \qquad (4)$$

Расчет показателя D_2 (4) выполняется для нескольких значений m, определяется точка насыщения, начиная с которой D_2 примерно постоянна: $D_2(m^*) = \text{const}$. Полученное значение (m*) принимается в качестве оценки размерности вложения.

Корреляционная размерность используется многими авторами в качестве количественной характеристики структуры аттрактора, в частности плотности его траекторий на отдельных участках.

Для аттракторов, реконструированных по биомедицинским сигналам, характерна большая размерность N (2). Обычно используется от нескольких десятков до сотен тысяч точек. В этом случае использование характеристики вида (3) и (4) приводит к большим временным затратам, что затрудняет обработку сигналов в реальном времени. Кроме того, по формулам (3) и (4) достаточно трудно локализовать участки аттрактора с высокой плотностью.

В работе [17] предложен более простой способ оценки плотности траекторий аттрактора. Для этого его двумерная проекция покрывается регулярной сеткой с шагом d = const. Определяется число точек аттрактора, попавших внутрь каждой j-й ячейки (h_j). Количество точек (r_j), оказавшихся на границе ячеек j-ой и j+1-й, делится поровну между граничными клетками. Отношение числа точек аттрактора, связанных с ячейкой (k_j) к ее площади (S_j) условно определяется как плотность аттрактора на j-м участке двумерной

проекции:

$$p_j = k_j / S_j \, , k_j = h_j + r_j / 2 \, . \quad (5)$$

Исследования характеристик аттракторов (4) и (5) выполнялись с использованием предварительно отфильтрованных от аппаратных шумов биомедицинских сигналов (продолжительностью 8 секунд).

В рамках данной работы особое внимание отведено рассмотрению влияния шума на точность оценки показателей аттрактора.

Мониторинг эмоций человека по ЭЭГ-сигналам

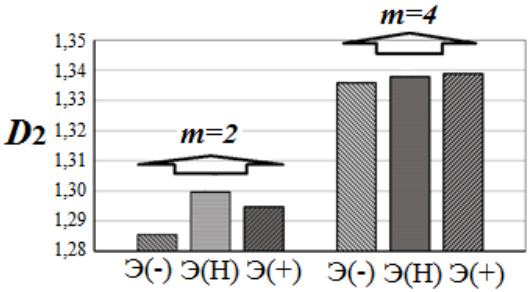
Рассмотрим мониторинг эмоциональных реакций человека на примере анализа ЭЭГ-сигналов. На полученные паттерны ЭЭГ-сигналов искусственно накладывались шумовые помехи, представляющие собой белый шум со ступенчато изменяющейся интенсивностью I, выраженной в процентах от максимального значения отсчета в генерируемом сигнале (от 10 до 40). В общей сложности сформировано 150 паттернов ЭЭГ с различным уровнем зашумленности на основе записей 10 испытуемых.

Для реконструкции аттрактора зашумленного сигнала в каждой записи выделен фрагмент временного ряда продолжительностью 2 000 отсчетов (N = 2000). Значение временного лага τ (2) рассчитывалось на основе автокорреляционной функции. Экспериментально установлено, что интенсивность шумовой компоненты не оказывает существенного влияния при выборе τ.

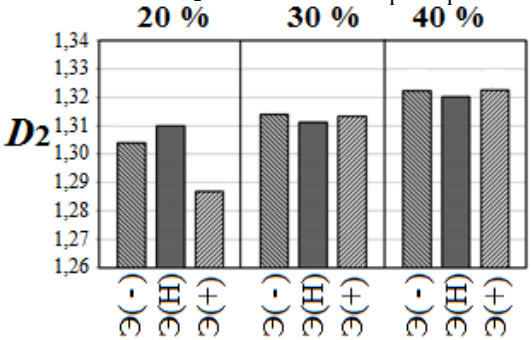
Полученные образцы оценивались по трем параметрам: величина корреляционной размерности восстановленного аттрактора D_2 (4), плотность траекторий двумерной проекции аттрактора в четырех центральных ячейках регулярной сетки p (5) и количество пустых ячеек k.

На рис. 1а представлен график изменения корреляционной размерности D_2 (4) (при $m = 2$ и $m = 4$) в зависимости от эмоции испытуемого (при $\tau = 15$ и $N = 2\,000$).

В ходе эксперимента установлено, что с повышением размерности вложения величина D_2 увеличивается, при этом практически выравниваются ее значения для различных знаков эмоций. При $m = 2$ средний диапазон изменения корреляционной размерности по всем участникам составил 0,015. При зашумлении паттерна ЭЭГ помехой, интенсивность которой изменялась ступенчато от 20% до 40%, при $m = 2$ наблюдается постепенное увеличение корреляционной размерности D_2 (рис. 1, б). Наиболее сильные изменения происходят при помехе 20%: D_2 для паттернов с отрицательным стимулом резко возрастает, а при положительном стимуле этот показатель увеличивается слабее, по сравнению с чистым ЭЭГ-сигналом. При увеличении помехи величина корреляционной размерности для всех паттернов ЭЭГ увеличивается: до 1,31 (при 30%), а затем до 1,32 (при 40%).



а – усредненный показатель D_2 в зависимости от размерности вложения m



б – усредненный показатель D_2 в зависимости от интенсивности шумовой компоненты I

Рис. 1. Изменение значений усредненного показателя D_2 (положительные эмоции – Э(+); отрицательные эмоции – Э(–); нейтральное состояние – Э(Н))

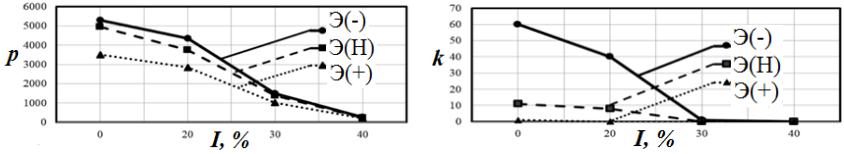
Проведенный анализ двумерной проекции аттрактора по двум параметрам ρ и k (5) показал, что с помощью этих показателей можно дифференцировать паттерны ЭЭГ по знаку эмоций, причем условие дифференциации будет иметь вид

$$p^н - p^+ > 0\,;\ k^н - k^+ > 0\,,\ \text{при положительных эмоциях,}$$
$$p^н - p^- < 0\,;\ k^н - k^- < 0\,,\ \text{при отрицательных эмоциях,} \qquad (6)$$

где $p^н$ – плотность траекторий аттрактора в центре сетки при нейтральном стимуле; p^+ и p^- – плотность траекторий при положительном и отрицательном стимуле соответственно; $k^н$ – количество пустых ячеек регулярной сетки при нейтральном стимуле; k^+ и k^- – количество пустых ячеек при положительном и отрицательном стимуле соответственно.

По мере увеличения интенсивности шумовой компоненты (I, %) данное соотношение сохраняется (рис. 2). Однако при шуме 30% показатель плотности p (5 и 6) в центре аттрактора практически перестает выступать в роли

дифференцирующего параметра значка эмоций. Аналогичная тенденция характерна и для показателя, иллюстрирующего количество пустых ячеек регулярной сетки *k*.



а б

Рис. 2. Плотность траекторий двумерной проекции аттрактора в центре (а) и количества пустых ячеек (б) в зависимости от интенсивности шума (m = 2 и τ = 15)

Заключение

Проведенные эксперименты показали, что широко используемый показатель корреляционной размерности D_2 (4) теряет свою чувствительность к изменению знака эмоциональных реакций с ростом зашумленности ЭЭГ-сигналов при использовании аттракторов с размерностью вложения m (2) больше двух.

При использовании плотности траекторий p (5) двумерной проекции аттрактора в центре регулярной сетки, а также с учетом количества пустых ячеек для классификации ЭЭГ-сигналов по знаку эмоций допустима интенсивность шумовой компоненты не более 30 %.

Дальнейшее направление исследований видится в дополнении модели эмоций, ориентированной на задачи интерпретации знака, уровня и динамики эмоциональных реакций человека по разнотипным биомедицинским сигналам (ЭЭГ, ЭМГ) новыми характеристиками на основе нечетких трендов, которые формируются по инвариантам нелинейной динамики, в том числе и по характеристикам аттракторов.

Список литературы

1. Рабинович М.И., Мюезинолу М.К. Нелинейная динамика мозга: эмоции и интеллектуальная деятельность // Успехи физических наук. 2010. Т. 180, № 4. С. 371–387.
2. Rangayyan R.M. Biomedical signal analysis: 2nd edition. New York: Wiley-IEEE Press, 2015. 720 p.
3. Lin J., Spraragen M., Zyda M. Computational models of emotion and cognition // Advances in Cognitive Systems. 2012. V. 2. P. 59–76.
4. Филатова Н.Н., Сидоров К.В. Компьютерные модели эмоций: построение и методы исследования. Тверь : ТвГТУ, 2017. 200 с.
5. Filatova N.N., Sidorov K.V., Terekhin S.A., Vinogradov G.P. The system for the study of the dynamics of human emotional response using fuzzy trends /A. Abraham et al. (Eds.) // Proceedings of the First International Scientific Conference “Intelligent Infor-

mation Technologies for Industry" (IITI'16), Advances in Intelligent Systems and Computing 451. Springer International Publishing Switzerland, 2016. Vol. 2, Part III. Pp. 175–184.
6. Filatova N.N., Sidorov K.V., Shemaev P.D., Rebrun I.A. Emotion and cognitive activity monitoring system // Proceedings of the 3rd Russian-Pacific Conference on Computer Technology and Applications "RPC 2018" (Russia, Vladivostok, August 18-25, 2018). IEEE, 2018. Pp. 166–169.
7. Jasper H.H. The ten-twenty electrode system of the international federation // Electroencephalography and Clinical Neurophysiology. 1958. Vol. 10. Pp. 371–375.
8. Fridlund A.J., Cacioppo J.T. Guidelines for human electromyographic research // Psychophysiology. 1986. Vol. 23, No. 5. Pp. 567–589.
9. Старченко И.Б., Перервенко Ю.С., Борисова О.С., Момот Т.В. Методы нелинейной динамики для биомедицинских приложений // Известия ЮФУ. Технические науки. 2010. № 9 (110). С. 42–51.
10. Lan Z., Sourina O., Wang L., Liu Y. Real-time EEG-based emotion monitoring using stable features // The Visual Computer. 2016. Vol. 32, No. 3. P. 347–358.
11. Filatova N.N., Sidorov K.V., Shemaev P.D. Prediction properties of attractors based on their fuzzy trend. In: A. Abraham et al. (Eds.) // Proceedings of the Second International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'17), Advances in Intelligent Systems and Computing 679. Springer International Publishing, 2018. Vol. 1. Pp. 244–253.
12. Филатова Н.Н., Сидоров К.В., Шемаев П.Д. Мониторинг характеристик аттракторов для оценки изменений эмоционального состояния человека // XIX Международная научно-техническая конференция «НЕЙРОИНФОРМАТИКА-2017» (02-06 октября 2017 г., г. Москва, Россия) : Сборник научных трудов в 2-х частях. Ч. 2. Москва : НИЯУ МИФИ, 2017. С. 48–57.
13. Takens F. Detecting strange attractors in turbulence // Dynamical Systems and Turbulence: Proceedings of a Symposium Held at the University of Warwick 1979-80 / ed. by D. Rand and L.-S. Young. Berlin, Germany: Springer Berlin Heidelberg, 1981. Vol. 898 of the series Lecture Notes in Mathematics. P. 366–381.
14. Grassberger P., Procaccia I. Measuring the strangeness of strange attractors // Physica D: Nonlinear Phenomena. 1983. Vol. 9, No. 1-2. Pp. 189–208.
15. Fraser A.M. Reconstructing attractors from scalar time series: A comparison of singular system and redundancy criteria // Physica D: Nonlinear Phenomena. 1989. Vol. 34, No. 3. P. 391–404.
16. Fraiser A.M., Swinney H.L. Independent coordinates for strange attractors from mutual information // Physical Review A. 1986. Vol. 33, No. 2. Pp. 1134–1140.
17. Filatova N.N., Sidorov K.V., Iliasov L.V. Automated system for analyzing and interpreting nonverbal information // International Journal of Applied Engineering Research. 2015. Vol. 10, No. 24. P. 45741–45749.

V. A. ANTONETS
Lobachevsky State University of Nizhny Novgorod
Institute of Applied Physics, Russian Academy of Sciences

SIMULATION OF INTUITIVE EVALUATION OF UNLIKE SEMANTIC OBJECTS

The main postulate of proposed semantically heterogeneous objects – comparison model is the comparison is meaningful for a subject only lying in the assumption that exchange-like transactions between subjects take place. The methodology employed can neither be attributed explicitly to normative nor to descriptive branches of decision-making science, because it doesn't relies on hypothesis as to how we make decisions and the collection of empirical data, but on psychophysiological characteristics of a human on the premise that it's important.

Keywords: intuitive thinking, stimuli perception, semantic differential, significance measurement, Pareto curve.

Introduction

The fact that inspired to write this work is: people are able to compare semantically heterogeneous objects. E.g., this is evidenced by the money we pay for goods and services, i.e. money for life values that constitute its quality. It's surprising about goods and services, that, in all their diversity, they have their objective price to range, yet the values are always subjective; moreover, we may vary the importance we attach to same value, but in different situations and over time: «A horse, a horse! My kingdom for a horse!»

Based on the above, the main postulate of this work is the comparison is meaningful for a subject only lying in the assumption that exchange-like transactions between subjects take place (the subject isn't required to be even a human – it may be a spirit, a Deity, etc.) At the same time, psychophysiological characteristics of a human – based approach is employed to simulate the model of decision-making when buying a good/service.

The research of decision-making processes is a relevant problem of psychology, neurophysiology, economics, computer sciences, politology, sociology, and other areas of expertise. Its complexity is related to the variety of cognitive mechanisms of decision-making and diversified aspects of research corresponding to the interests of different sciences.

When considering a decision-making process, two types of solutions are distinguished, namely the solutions based on intuitive evaluation and those based on logical arguments [1]. Logical evaluations require substantial mental efforts and time. In spite of it, owing to incomplete or inaccurate data, logical evaluations do not often correspond to reality. In contrast to rational solutions, intuitive solutions

are taken spontaneously without any efforts and do not take a long time. It is quite natural that the character of these solutions always depends on subjective perception of the situation. However, it does not mean that the mechanism of intuitive decision-making cannot be quantitatively described.

One of the most important modern scientific results in the field of studying the mechanisms of making intuitive decisions [1] lies in the demonstration of the fact that an intuitive evaluation is formed under the effect of object features that are the most accessible for perception, but which, however, are not always the most significant for the evaluation and making a choice from the point of view of objectiveness (rationality). At that, it is noted that regularities of intuitive thinking have a profound similarity with the laws of sensory perception. Taking into account that it is possible to study and quantitatively describe sensory perception laws [2, 3], there arises a natural question concerning a possibility to quantitatively describe regularities of evaluation and decision-making regarding unlike objects, in particular goods, each of them being characterized by a unique set of unlike semantic features. Since the principles of decision-making are usually studied as a system of conclusions that is simulated at a purely qualitative level, the problem of a possible quantitative description of intuitive evaluation of unlike semantic objects in the process of decision-making is new and, undoubtedly, rather interesting.

Types of stimuli and principles of their evaluation. Objects of intuitive evaluation

A possibility in principle to quantitatively describe human perception of sensory stimuli was first discovered by Bouguer [4] when studying perception of brightness of luminous objects. As far as stimuli of other modalities are concerned, such possibility was demonstrated more than a century ago by Herman Helmholtz [5, 6], Ernest Heinrich Weber and Gustav Theodor Fechner [7, 8]. They were the first to construct quantitative psychophysical sensory scales describing a relationship between stimuli and perception intensities. By the middle of the XX century their investigations had been expanded and specified. First, it was verified that a man really has a unique ability to perform a quantitative comparison and to evaluate his perceptions under the action of stimuli of different modalities. Moreover, a man is capable of comparing the intensity of perception induced by stimuli of different physical nature, i.e., is capable of intermodal comparison [3]. Thanks to this ability it is possible to introduce a quantitative measure of the intensity of subjective perceptions. Second, it was shown by the example of the stimuli of the key modalities that a quantitative dependence of the perception intensity on that of the stimulus that has induced it can be described by the generalized Weber—Fechner—Stevens law [2] the differential form of which is described by the equation:

$$\frac{\Delta S}{S} = k\frac{\Delta R}{R}, \qquad (1)$$

here:
– ΔS is the subjective evaluation of the variations in the perception intensity S;
– ΔR is the variation in the stimulus R which caused changes in the perception intensity ΔS;
– $k > 0$ is the constant depending on the perception modality.

The integral form of Eq. (1) is as follows:

$$\frac{S}{S_0} = \left(\frac{R}{R_0}\right)^k, \qquad (2)$$

here: S and S_0 are the compared subjective perception intensities formed under the action of the stimuli R and R_0, respectively.

A substantial limitation of the application of psychophysical scales is as follows: they are actually applicable for evaluating perception intensity of purely physical stimuli that are unambiguously characterized by positive scalar quantities. Thus, psychophysical scales cannot be applied in case of the stimuli the features of which are semantic rather than sensory, i.e., they are meaningful and have a negative, neutral, or positive coloring. Nevertheless, we often have to evaluate the objects characterized by a complex set of features. In particular, a product is a complex semantic stimulus since its quality is characterized by a whole set of semantically unlike features that are significant for a consumer.

An example of intuitive evaluation of objects characterized by a complex totality of features is refereeing of such kinds of sports as figure skating, gymnastics, diving, etc. The evaluation process in the above examples lies in reducing an intuitive impression on a complex object to a unique value, namely a product price or a point for the performed exercise. At that, if an evaluation is made by an expert or a referee who is disinterested in the outcome, he is indifferent to the result to be obtained. If a product is evaluated by a buyer, we cannot expect indifference from him since he is to make a decision whether to buy it or not. There are many factors which influence the buyer's decision-making on purchasing a product. Due to these factors, a buyer evaluates a product as an interested person rather than an objective expert. These factors can include different limitations (time, financial, etc.) as well as views and principles a person is guided by when making a decision. These views are often associated with intuition, can be based on previous experience, and represent themselves hypotheses that cannot be proved. Thus, a buyer's evaluation of a product or a totality of products that are necessary for his life even in case it is made with the participation of experts (advisers and consultants) is, undoubtedly, subjective and often intuitive. In the present paper, we demonstrate a solution of the problem of quantitative description of intuitive evaluation of an individual product or a product basket using the method of semantic differential and complex semantic stimuli significance measurement.

Method of semantic differential. Quantitative description of a subjective evaluation of product quality

An attempt of a formalized description of man's attitude to unlike objects was undertaken by an American psychologist Charles Osgood as early as in the middle of the XX century. In the result, he revealed and demonstrated a human ability to quantitatively describe his attitude to the object to be evaluated through a system of bipolar features the expressiveness of which is measured with the help of discreet and even continuous scales [9]. It is of fundamental importance that these scales contain a zero point that corresponds to a neutral value of an individual's perception of particular feature expressiveness. Thanks to the existing zero, these scales are the scales of attitudes [2], i.e., make it possible, based on the intensity of an associative reaction induced by them, not only to organize the objects (order scales) and to define distance intervals between them (interval scales), but also to determine the significance of each object with respect to some fundamental reference object (attitude scales). Their totality was called by Ch. Osgood a semantic differential [9]. Based on a set of values obtained using the significance scales an evaluated object can be represented as a multidimensional vector in the space of semantic feature possessing a metric. Thus, it becomes possible to study a mutual positioning of objects in this space and to measure a distance between them.

At present, the procedures implementing the method of semantic differential are widely used in the areas where it is necessary to quantitatively describe an individual subjective attitude of an expert to some object properties or some aspect of an event or a situation. In particular, they are used for integral comparison of identical products for their positioning in a market segment [10].

For a quality of the considered product to be quantitatively expressed, it is necessary first to list its properties that are significant for a consumer and the totality of which characterizes this product. After that, each i-th property is ascribed a weight w_i in accordance with the consumer's opinion. Thus, a vector of significance of product characteristics $\mathbf{w}$ normalized so that $\|\mathbf{w}\| = 1$ is formed. Then, according to the consumer's opinion which can be either intuitive or based on the measurement or argumentation, a vector of property expressiveness $\mathbf{v}$ is formed for the considered product. A degree of expressiveness of a particular product property as compared to an ideal one is determined with the help of Ch. Osgood's scales the marks of which reflect an individual opinion of a consumer on the evaluated object. The integral value q of the quality of the considered product is expressed through a scalar product of the vectors $\mathbf{v}$ and $\mathbf{w}$:

$$q = (\mathbf{v},\mathbf{w}). \qquad (3)$$

Based on the characteristics of the evaluated product that are considered the most significant by a consumer we can judge on the properties an "ideal" product should possess from his point of view. Since an "ideal" object is determined based on the significance of its properties, the integral evaluation of its quality in case

the values of the properties for the evaluated product coincide with the corresponding properties of the "ideal" object, i.e., **v** = **w**, is $q = (\mathbf{w},\mathbf{w}) \equiv 1$. For a nonideal product, $q < 1$.

Evaluation of a subjective consumer reaction to a product

While making a decision to buy a consumer compares not only the significance of the product to be bought but also acceptability of its quality and price. It means that there is a dependence of the acceptable price of an individual product on its quality q, i.e., $p = p(q)$. When buying a product, a consumer perceives it as a stimulus and reacts to it via an evaluation of its quality and the amount of resources he is ready to spend for buying it.

Consider the case when money serve as a resource c, and a reaction is expressed by the amount of money a consumer is ready to pay for the product. If we assume that this reaction obeys the Stevens law, it can be described by relation (1) which will take the form:

$$\frac{\Delta q}{q} = k\frac{\Delta c}{c}. \tag{4}$$

In the integral form, Eq. (4) transforms into

$$\left(\frac{q}{q_0}\right)^{-k} = \left(\frac{c}{c_0}\right). \tag{5}$$

Let us determine the parameters q_0 and c_0 from the following considerations. Let c_0 be the maximal resource (or the maximal price) $p_{\max}$ a consumer is ready to spend for buying this product based on some subjective considerations (evaluation of the need satisfied by the product, the idea of a reasonable price, etc.). Let p be his subjective evaluation of the price for the product with the quality value of q. Then, $c_0 = p_{\max}$, and $c = p_{\max} - p$. In this case, Eq. (5) transforms into Eq. (6):

$$p = p_{\max} \cdot \left[1 - \left(\frac{q}{q_0}\right)^{-k}\right]. \tag{6}$$

There is a threshold product quality $q = q_{\min}$ for each consumer, lesser of which he will not accept even free of charge since such product is not suitable for the implementation of his need. In this case, $p = 0$ and, thus, $q_0 = q_{\min}$. So Eq. (6) finally transforms into:

$$p = p_{\max} \cdot \left[1 - \left(\frac{q}{q_{\min}}\right)^{-k}\right]. \tag{7}$$

Here: p is the product price, q is the integral evaluation of its properties, $k > 0$ is the constant.

The obtained expression (7) represents itself a well-known Pareto formula represented by the curve in Fig. 1.

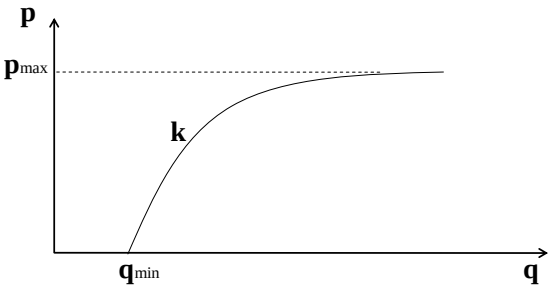


Fig. 1. “Price—Quality” characteristic curve

Thus, the hypothesis on the similarity of mechanisms of intuitive thinking and sensory perception formulated in [1] and extended by us up to the supposition on applicability of the Stevens law [2] for description of an intuitive reaction to complex semantic stimuli yields a reasonable quantitative description of a subjective reaction of a buyer on an individual product characterized by a totality of semantically unlike features expressed through a “price—quality” relationship.

Consumer’s unlike products basket evaluation model

As it was noted above, there is a number of factors based on which a consumer evaluates a product as an interested person rather than as an objective expert, and his decision on buying is in the majority of cased subjective. However, this fact does not mean that a consumer’s behavior in the process of selecting a limited set of products cannot be quantitatively described. To solve the stated problem, let us use the above results of evaluating an individual product.

According to modern marketing principles, let us assume that each type of manufactured products is intended for satisfying a certain need or demand [12]. Owing diversified needs and demands as well as methods and technologies of their satisfaction, an assortment of product types inevitably turns out to be semantically unlike.

Let us assume that the number of needs and demands which are taken into account by a consumer at a certain instant is limited. In this case, each individual need or demand to be satisfied is characterized by and differs from other relevant needs and demands by its acuity. It allows us to think that a consumer has an idea about the assortment (basket) of goods he currently needs, and all the products in his mind are organized according to a degree of acuity of a corresponding need. At that, the product relevance and the acuity of the need to buy it are identical.

Thus, according to the aforementioned results and assumptions, in the process of describing the logic of evaluating a basket of unlike goods we will proceed from the assumption that when making a decision on buying a product a consumer is capable of:

- distinguishing the products that are the most significant for him;
- to organize the products in his mind according to a degree of acuity of the needs to be satisfied by them;
- to evaluate the integral quality q_i of each individual i-th product (i.e., a degree of its suitability for satisfying a corresponding need, based on the perception of a complex set of unlike semantic feature characterizing the product).

To formalize the procedure of arranging products according to the degree of need acuity corresponding to them, let us introduce a normalized vector of the significance of needs $\mathbf{r} = (r_1, r_2, \ldots r_k)$, where r_i is the acuity which characterizes for him the significance of the *i-th* product. Since this vector depends only on the consumer's ideas it is expedient to normalize it so that $\|\mathbf{r}\| = 1$. At that, it is important to note that contrary to the vector **r** the vector **q** cannot be a priori normalized by 1. It is related to the fact that q_i depends not only on a consumer's subjective ideas concerning the reasonable quality of the product he would find acceptable but also on the evaluation of the properties of the actual offered product which can both fail his expectations or exceed them.

The degree of a consumer's satisfaction with each individual *i-th* product its determined by its suitability for meeting the *i-th* need and depends on a whole set of its qualities. We assume that general satisfaction with a basket is determined by the acuity of needs and a quality of their satisfaction. Let us introduce a scalar product *S* of the vectors **r** and **q** as a quantity characterizing a degree of a consumer's satisfaction with the product basket:

$$S = (r, q) = \sum r_i \cdot q_i \ . \tag{8}$$

For each consumer, it is important to buy such a set of products for which the degree of his satisfaction, which we will characterize by the quantity *S*, is maximal at a given total price p of the whole product basket. This condition can be described as follows:

$$\frac{S}{p} = \frac{\sum r_i \cdot q_i}{\sum p_i(q_i)} \to \max . \tag{9}$$

Here: p_i is the price of the *i-th* product, q_i is the integral evaluation of its quality, and p is the total price of the bought products.

Here, it is necessary to mention that a list of needs and an assortment of the bough product basket can be non-coinciding since, as it has already been mentioned, there is a minimal threshold quality q_{min}, which is individual to each consumer. When $q_i < q_{i\,min}$, one should refuse to purchase it. Sometimes, when buying a product basket the situation is such that a consumer spontaneously buys a product that he was not planning to buy.

There exist many situations and principles of buying products from the basket and, as it was already mentioned, an acceptable price of the evaluated set of products depends on the views and principles a consumer is guided by when making a purchasing decision. Thus, e.g., in one case the product price expected by a buyer

can depend on his personal idea about a reasonable price for a product of such type and such quality. In another case, by comparing the product significance and quality he can base decisions on the existing financial limitations and reserve certain amounts of money for buying other products, i.e., for satisfying the rest of the needs. In this case, in the process of product basket formation the logic of a consumer's evaluation of products as unlike semantic objects turns out to be different.

Consider the case when, based on financial limitations, a consumer buys a set of products at a fixed sum c = const. In this case, $\sum p_i(q_i) = c$, and this task of Eq (9) extremum determination is added up optimization problem with limitation, which can be solved with the help of Lagrange method [13]. The extremum conditions are determined from Eq. (10):

$$\frac{\sum r_i \cdot q_i}{\sum p_i(q_i)} + \lambda\left(\sum_i p_i(q_i) - c\right) = 0, \qquad (10)$$

here: λ – is the constant (Lagrange multiplier).

It follows from (10) that $\frac{dp_i}{dq_i} \Big/ r_i$ does not depend on i. It means that, in its turn, the vectors **r** and $\text{grad}_q p$ at the point of the maximum of S/c are parallel.

Thus, the considered example of product basket formation demonstrates a possibility of quantitative description of the logic of evaluating a set of semantically unlike objects.

Conclusions

The proposed model of decision making on buying a product demonstrates the possibility of quantitative description of integral intuitive evaluation mechanism for qualitatively unlike objects. The obtained quantitative description of an intuitive reaction of a man to objects is based on the results of the studies by S. Stevens, D. Kahneman, and Ch. Osgood.

A quantitative model of the logic of evaluating semantically unlike objects is also demonstrated for the case of a subjective selection of a totality of such objects from a limited set. The logic of such evaluation is based on a quantitative expression of the degree of a consumer's satisfaction with the selected set of products in the form of a scalar product.

The significance of quantitative models of such type is related to the fact that by studying the characteristics of statistical distribution of consumers' perception of their needs and demands as well as the products offered to them on the market it will be possible to simulate market segment parameters. In its turn, it will make it possible to form consumers' ideas on the offered products and services and formulate more adequate requirements for them.

In this article two quantitative models of the mechanisms of human intuitive decision making when buying a product (goods) are constructed. The first model

demonstrates a possibility of quantitative description of a subjective evaluation of a product as a complex semantic object. The second model, constructed by the example of forming a product basket from the offered range of products, quantitatively describes a mechanism of comparison and subjective evaluation of a group of semantically unlike objects. The models have been constructed based on the results of Ch. Osgood, S. Stevens, and D. Kahneman.

Acknowledgement

The author are grateful to, P. V. Antonets, M. A. Antonets and A. A. Kharitonov for their assistance in preparation of the paper.

References

1. Kahneman D., Frederisk Sh. Representativeness Revisited, Attribute Substitution in Intuitive Judement// Heuristics and biases: The psychology of intuitive thought/ Ed. By Thomas Gilovich, Dale Griffin, and Daniel Kahneman. N.Y.: Cambridge Univ. Press, 2002, P. 49-81.
2. Stevens S.S. On the psychophysical law. // Psychol. Rev., 1957, 64, 153–181
3. Stevens S.S. Cross-modality validation of subjective scales for loudness, vibration and electric shock. // J. exp. Psychol., 1959, 201–209.
4. Bouguer P., Sur la meilleure méthode pour observer l'altitude des étoiles en mer et Sur la meilleure méthode pour observer la variation de la boussole en mer // Paris, 1727.
5. Helmholtz H. von, Handbuch der Physiologischen Optik. Bd. 3. Hamburg und Leipzig, Verlag von Leopold Voss, 1910.
6. Hermann L. F. Helmholtz, M. D., On the Sensations of Tone as a Physiological Basis for the Theory of Music (Fourth ed.) // Longmans, Green, and Co., 1912, 575 p.
7. Fechner G. Elemente der Psychophysik. 2 Bände, Leipzig, 1860,
8. Fechner G. Revision der Hauptpuncte der Psychophysik, Leipzig, 1882
9. Osgood C. E., Suci G., and P. Tannenbaum. The Measurement of Meaning // University of Illinois Press, 1957.
10. Viardot E. Successful Marketing Strategy for High-Tech Firms // 3rd edition. Norwood. MA: Artech House. Inc., 2004.
11. Pareto V. Manual of Political Economy // Trans. Ann S. Schwier, New York : Augustus M. Kelley, 1971.
12. Lamben J. Strategic marketing. European perspective // St. Peterburg : Nauka, 1996.
13. Bertsekas, Dimitri P. Nonlinear Programming (Second ed.) // Cambridge, MA. : Athena Scientific., 1999.

Н. Ш. АЛЕКСАНДРОВА[1], В. А. АНТОНЕЦ[2,3], И. В. НУЙДЕЛЬ[2,3], О. В. ШЕМАГИНА[2,3], В. Г. ЯХНО[2,3]

[1]Sprachbrücke E.V., Берлин, Германия
[2]Нижегородский государственный университет им. Н.И. Лобачевского
[3]Федеральный исследовательский центр Институт прикладной физики Российской академии наук, Нижний Новгород
yakhno@appl.sci-nnov.ru

МОДЕЛИРОВАНИЕ РЯДА ОСОБЕННОСТЕЙ ФОРМИРОВАНИЯ ЕСТЕСТВЕННОГО БИЛИНГВИЗМА*

Рассмотрены варианты динамических режимов обучения детей двум языкам на основе предложенной математической модели. Показано соответствие модельных режимов экспериментальным данным.

Ключевые слова: спонтанное обучение языкам, феноменологическая модель билингвизма, анализ релаксационных динамических режимов.

1. Введение

1.1. В работах [1–13] было показано, что естественное освоение детьми нескольких языков в языковой среде характеризуется следующими особенностями:

- В раннем возрасте (до трех лет) при двуязычном воспитании дети с одним доминирующим языком приближаются в этом языке к уровню монолингвов, но значительно отстают в другом языке. Сбалансированные же билингвы (вход каждого из языков примерно равен по времени) в обоих языках находятся «посередине», заметно отставая от уровня монолингвов [1]. Общий словарь монолингва предстает как максимум, которого можно достичь в этом возрасте.
- Сходная закономерность просматривается на основе наблюдений [2]: доминирование одного из языков при переходе к фразовой речи более благоприятно, чем при сбалансированном двуязычии.
- У билингвов наблюдается общая тенденция «уменьшения» каждого из освоенных языков [3–6]. В детстве и более старшем возрасте они обладают меньшим пассивным словарем и хуже оперируют вербальными стимулами, чем монолингвы.
- Естественный билингвизм возникает лишь при длительно сохраняющейся потребности общаться на обоих языках. Двуязычная среда является только

* Данная работа выполнена при поддержке Министерства образования и науки РФ, проект №14.Y26.31.0022

необходимым условием становления билингвизма, но не его причиной. Ребенок может не освоить второй язык, несмотря на то, что слышит его ежедневно [7,8].

- При естественном билингвизме языки могут не только развиваться и совершенствоваться в течение жизни, но также могут останавливаться в развитии (фоссилизации), обедняться и даже подвергаться аттриции – полному непатологическому стиранию. В одноязычной среде языковой регресс возможен лишь при тяжелой болезни [9–13].
- В популяции здоровых детей раннее многоязычие не всегда сочетается с высоким интеллектом и успешностью обучения в школе. Возможны также затруднения при изучении иностранных языков логическим способом [7].

1.2. В предлагаемой работе рассматривается модель динамических процессов, обеспечивающих освоение языка. Она включает три основных постулата:

- освоение языка является неотъемлемой частью общего процесса развития ребенка, а объем освоенного языка – индикатором развития ребенка;
- развитие ребенка происходит путем спонтанного (непроизвольного) формирования набора алгоритмов для осуществления операций над образами природной среды, в которую он погружен. Такой набор алгоритмов обычно называют «спонтанным знанием» как основы мировоззрения;
- происходит постоянная утрата (разрушение) части мало востребованных алгоритмов для реализации ранее освоенных операций, например, за счет их перевода из активной рабочей памяти в пассивное «хранилище».

2. Описание феноменологической модели

Для концептуального описания и анализа развития субъекта в условиях предоставленной ему среды и сопутствующего этому развитию спонтанному (бессознательному) овладению двумя языками как инструментами общения используем известный физический подход составления балансных уравнений (см., например, [14–17]).

Из экспериментальных данных и соображений здравого смысла при моделировании двуязычия ограничимся рассмотрением пяти основных (взаимосвязанных) переменных и трех важных параметров:

$M(t)$ – количество активных алгоритмов операций над образами окружающей среды. Такой набор алгоритмов обычно называется *мировоззрением субъекта*. Для каждого индивидуума $M \in (0; M_{max})$, возможно предопределяемом генотипом и средой;

$П(t)$ – поток внешних или внутренних сигналов, которые осознаются или ожидаются субъектом с мировоззрением $M(t)$;

$V_1(t)$, $V_2(t)$ – количество инструментальных алгоритмов (когнитивных фильтров), нашедших отображение в 1-м и 2-м языках соответственно;

$R(t)$ – пороговая переменная, определяющая переключение динамической системы освоения языков при активизации использования одного из них;

Э(t) – параметр эмоционального состояния субъекта, лежащий в интервале $0 < Э(t) < 1$, т.е. от наиболее "вдохновлённо-благоприятного" (вблизи от 0) до наиболее "разрушительно-огорчительного" (вблизи 1);

$E(t)$ – параметр энергетического состояния субъекта лежащий в интервале $0 < E(t) < 1$, т.е. от наиболее "истощенного" (вблизи от 0) до наиболее "эйфорично-уверенного" в своей силе (вблизи от 1).

$Q(t)$ – параметр стресса субъекта, лежащий в интервале $0 < Q(t) < 1$, изменяющийся от этапа активации к этапу закрепления решений, переходах к расслаблению, отдыху и "нормальному" состоянию.

Рассмотрим версии балансных уравнений, с помощью которых можно изучать динамические процессы при развитии субъекта.

2.1. Первое из предлагаемых балансных уравнений описывает изменение во времени количества активных алгоритмов операций над образами окружающей среды:

$$\tau_{1M}\frac{dM}{dt} = -\frac{\tau_{1M}}{\tau_{2M}}\cdot M + \left(1 + K_{p_1}(t)\cdot V_1 + K_{p_2}(t)\cdot V_2\right)\cdot F_M[-T_M + П], \qquad (1)$$

где: τ_{1M} – характерное время спонтанного обучения новым алгоритмам для работы с образами внешней среды (обычно, минуты - часы); τ_{2M} – характерное время забывания ранее воспринятых алгоритмов для работы с образами внешней среды (обычно, месяцы-годы); $K_{p1,}\, K_{p2}$ – весовые коэффициенты поступления информационных потоков по языковым каналам V_1 , V_2 соответственно; F_M – пороговая функция с параметром T_M.

2.2. Второе балансное уравнение описывает изменение потока сигналов П(t) (внешних или внутренних), которые осознаются или ожидаются субъектом, имеющим мировоззрение, характеризующееся объемом $M(t)$. П(t) пропорционально числу алгоритмов, которые система научилась генерировать для выделения из сенсорного потока важных для нее элементов.

$$\tau_{1П}\frac{dП}{dt} = -\frac{\tau_{1П}}{\tau_{2П}}\cdot П + F_П\left[-T_П + \sigma_{ПМ}\cdot M + \sigma_{П_1}\cdot V_1 + \sigma_{П_2}\cdot V_2\right], \qquad (2)$$

где: $\tau_{1П}$ – характерное время изменения потока осознаваемых внешних сигналов (обычно, секунды - минуты); $\tau_{2П}$ – характерное время забывания или разрушения ранее воспринятых внешних сигналов (обычно, часы – дни - месяцы); $F_П$ – пороговая функция с параметром $T_П$; $\sigma_{ПМ}$, $\sigma_{П1}$, $\sigma_{П2}$ – весовые коэффициенты, определяющие влияние $M(t)$, V_1 , V_2 соответственно на активацию осознания воспринимаемых сигналов.

2.3. Третье балансное уравнение описывает изменение числа активных инструментальных алгоритмов и "когнитивных фильтров" для освоения одного из языков общения.

$$\tau_{1V_i}\frac{dV_i}{dt} = -\frac{\tau_{1V_i}}{\tau_{2V_i}}\cdot V_i + F_i\left[-T_i(\mathrm{M},\Theta,Q,E) + \Pi + \gamma_{ii}\cdot V_i - \sum_{j\neq i}\gamma_{ij}\cdot V_j\right] \quad (3)$$

где: τ_{1Vi} – характеное время освоения; τ_{2Vi} – характеное время забывания; F_i – пороговая функция с параметром $T_i(\mathrm{M},\Theta,Q,\mathrm{E})$; γ_{ij} – весовые коэффициенты, определяющие влияние *j*-го языка на активацию изучения *i*-го языка. $T_i(\mathrm{M},\Theta,Q,\mathrm{E})$ – порог запуска алгоритмов обучения *i*-той области деятельности, в частности, языку общения. Для описания освоения языков в двуязычной среде *i* принимает значения 1 или 2.

Из соображений здравого смысла можно использовать, например, зависимость: $T_i(\mathrm{M},\Theta,Q,\mathrm{E}) = T_{i0} - a_{iM}\cdot\mathrm{M} + a_{i\Theta}\cdot\Theta + a_{iQ}\cdot Q + a_{iE}\cdot(1-E)$.

Для простоты описания и анализа режимов обучения рассмотрим случай обучения только двум языкам (билингвизм). Соответственно, получим систему из двух уравнений:

$$\tau_{1V_1}\frac{dV_1}{dt} = -\frac{\tau_{1V_1}}{\tau_{2V_1}}\cdot V_1 + F_0\left[-T_1 + \gamma_{11}\cdot V_1 - \gamma_{12}\cdot V_2\right] \quad (4)$$

где τ_{1V_1} – характерное время спонтанного обучения новым элементам первого языка (обычно, дни – недели); τ_{2V_1} – характерное время разрушения доли элементов первого языка или перевод их из активной рабочей памяти в пассивное "хранилище" (обычно, месяцы – годы).

2.4. Четвертое балансное уравнение описывает изменение числа активных инструментальных алгоритмов и "когнитивных фильтров" для освоения второго языка:

$$\tau_{2V_2}\frac{dV_2}{dt} = -\frac{\tau_{1V_2}}{\tau_{2V_2}}\cdot V_2 + F_0\left[-T_2 - \gamma_{21}\cdot V_1 + \gamma_{22}\cdot V_2\right], \quad (5)$$

где τ_{1V_2} – характерное время спонтанного обучения новым элементам второго языка (обычно, дни – недели); τ_{2V_2} – характерное время разрушения доли элементов второго языка или перевод их из активной рабочей памяти в пассивное "хранилище" (обычно, месяцы – годы).

Нелинейные функции в приведенных выше уравнениях (1) – (2) и (4) – (5) имеют ступенчатую зависимость. Для упрощения начального рассмотрения используем для каждой из них единичную функцию Хевисайда: $F_M[\ ] = F_\Pi[\ ] = F_{V_i}[\ ] = F_0[\ ]$.

2.5. Пятое уравнение определяет последовательность операций $R_i(t)$ (6), которая выполняется распознающей системой для определения наиболее важных черт внешней ситуации и условий, в которых необходимо использо-

вать тот или иной язык (адекватный ситуации) для общения и включения алгоритмов обучения, соответствующих именно этому языку. За время τ_p (доли секунды) происходит распознавание и переключение:

а) для первого языка использование только уравнения 4, при постоянной величине V_2 и $K_{p_1}(t) = K_{0\,p_1}$, $K_{p_2}(t) = 0$;

б) для второго языка использование только уравнения 5 при постоянной величине V_1, и $K_{p_1}(t) = 0$, $K_{p_2}(t) = K_{0\,p_2}$. (6)

2.6. Из известных экспериментальных данных времена реагирования основных переменных системы и важных управляющих параметров располагаются в следующем порядке:

τ_p <<	$\tau_П$ < или <<	$\tau_М$ < или <<	$\tau_Э$ <	$\tau_Е$ <	τ_Q < или <<	τ_{1Vi} ≈	τ_{2Vi}
доли секунды	сек. - мин.	мин. - часы	мин.-часы - дни	мин.-часы-недели	мин.-часы-недели	дни - недели	дни - недели

(7)

Таким образом, при рассмотрении динамических процессов можно выделить иерархию этапов "быстрых" и "медленных" изменений переменных. Важные параметры, связанные с эмоциональными, энергетическими и стрессовыми процессами в рамках данного рассмотрения будем считать постоянными.

3. Рассмотрение ряда режимов функционирования феноменологической модели

Анализ нелинейных уравнений (1)–(2) и (4)–(5) и описываемых ими динамических процессов проводился с помощью стандартных радиофизических методов анализа релаксационных систем [14–16].

3.1. Рассмотрим фазовые плоскости для описания процессов изменения М и П, уравнения (1) и (2), с учетом величин "обученности" для первого и второго языков. При этом все остальные переменные считаем постоянными. Фазовые портреты этой системы приведены на рис. 1-2. Видно, что для малых или отрицательных порогов $T_П$ и $T_М$ система генерирует новые алгоритмы для выделения новых сигналов – образов из внешнего сенсорного потока (рис. 1). Соответственно увеличивается и М - число алгоритмов для выполнения операций с образами внешней среды ($М \to М_1$ или $М_2$, а $П \to \frac{\tau_{2П}}{\tau_{1П}}$).

Таким образом, из уравнений (1) и (2) следует, что большие пороги включения механизмов обучения и осознанного восприятия потока внешних сенсорных сигналов могут блокировать спонтанное обучение. Известно, что эти пороги возрастают при неблагоприятных эмоциональных состояниях, а также недостаточном энергетическом обеспечении субъекта.

3.2. Фазовые плоскости для описания процессов изменения обучения двум языкам рассмотрим в предположении, что М и П достигли своих стационарных состояний:

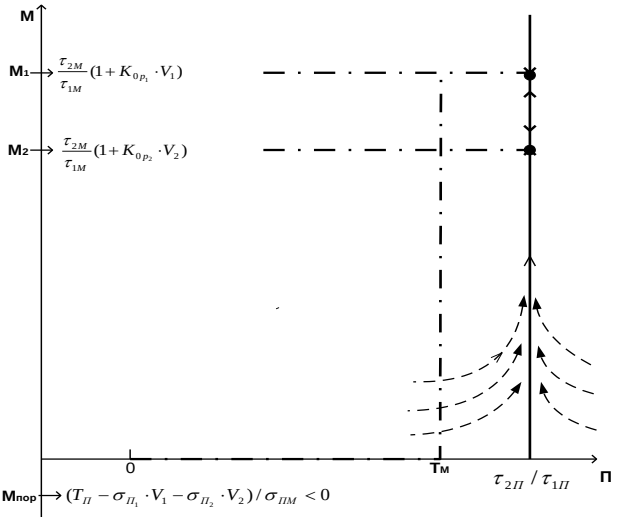
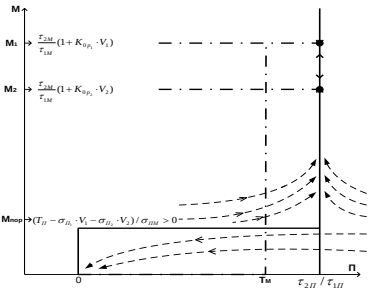
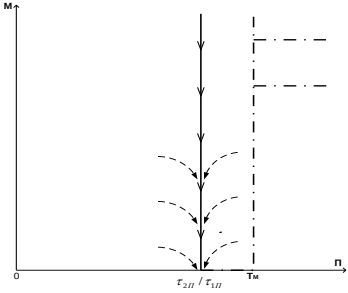


Рис. 1. Движение изображающей точки в случае $T_{\text{М}} < \frac{\tau_{2\text{П}}}{\tau_{1\text{П}}}$ и $\frac{\left(T_{\text{П}} - \sigma_{\text{П}_1} \cdot V_1 - \sigma_{\text{П}_2} \cdot V_2\right)}{\sigma_{\text{ПМ}}} < 0$



а)



б)

Рис. 2. Движения изображающих точек: а) в случае $T_{\text{М}} < \frac{\tau_{2\text{П}}}{\tau_{1\text{П}}}$ и $\frac{\left(T_{\text{П}} - \sigma_{\text{П}_1} \cdot V_1 - \sigma_{\text{П}_2} \cdot V_2\right)}{\sigma_{\text{ПМ}}} > 0$; б) в случае $T_{\text{М}} > \frac{\tau_{2\text{П}}}{\tau_{1\text{П}}}$ и $\frac{\left(T_{\text{П}} - \sigma_{\text{П}_1} \cdot V_1 - \sigma_{\text{П}_2} \cdot V_2\right)}{\sigma_{\text{ПМ}}} < 0$

Увеличение порогов $T_{\text{П}}$ и $T_{\text{М}}$ приводит к появлению зоны разрушения ранее созданных алгоритмов анализа сигналов из внешней среды (см. на рис.

2а область $М < М_{пор}$, где $М \to 0$ и $П \to 0$), либо блокировку только мировоззренческих алгоритмов ($М \to 0$) при росте осознания сенсорных сигналов $П \to \frac{\tau_{2П}}{\tau_{1П}}$ (см. рис. 2б).

$M = \frac{\tau_{2M}}{\tau_{1M}} \cdot \left(1 + K_{0p_1} \cdot V_1\right)$ для первого языка; $M = \frac{\tau_{2M}}{\tau_{1M}} \cdot \left(1 + K_{0p_2} \cdot V_2\right)$ для второго языка; $П = \frac{\tau_{2П}}{\tau_{1П}}$. Эти значения подставим в уравнения (4) и (5). В этом случае пороги T_1 и T_2 могут стать отрицательными, что обеспечивает условия спонтанного обучения языкам общения. Анализ возможных динамических режимов изменения переменных V_1 и V_2 в последовательностях этапов обучения то первому, то второму языку наглядно показан на фазовых плоскостях на Рис. 3. Неустойчивые ветки нуль – изоклин обозначены как "порог 1" для $dV_1/dt = 0$ и "порог 2" для $dV_2/dt = 0$.

Важно отметить, что в случае обучения только одному языку (например, V_1, траектория **a** –> **k** на рис. 3а) величина переменной будет превышать величину каждой из переменных V_1 и V_2, полученных на этапах их последовательного обучения. Если число и длительности этапов обучения для V_1 и V_2 суммарно равны этапам обучения только для V_1, то V_2 для монолингва будет в два раза превышать величины V_1 и V_2 для билингва.

Примеры динамических процессов (траекторий) при попытках обучения двум языкам приведены также на рис. 3б.

Видно, что в демонстрируемых случаях траектории изменения переменных зависят от выбора начальных состояний переменных. При этом на рисунке можно выделить четыре области [(1); (2); (3); (4)] с разными особенностями развития процессов. Аналогичный анализ возможных траекторий движения изображающей точки был получен и для случая T_1 и T_2 больше нуля.

Для проверки адекватности полученных модельных результатов требуется обсуждение с практическими психологами возможностей использования имеющихся методик или разработка новых подходов для измерения в реальных условиях основных параметров рассмотренной модели (например: $\tau_{1М}, \tau_{2М}, \tau_{1П}, \tau_{2П}, T_М, T_П, \tau_{1V_i}, \tau_{2V_i}, T_{V_i}, \gamma_{ij}$ и других определяющих параметров). При получении таких данных возможна не только верификация используемого модельного рассмотрения, но и на его основе формирование методик коррекции регистрируемых нарушений в наблюдаемых случаях спонтанного, естественного обучения детей.

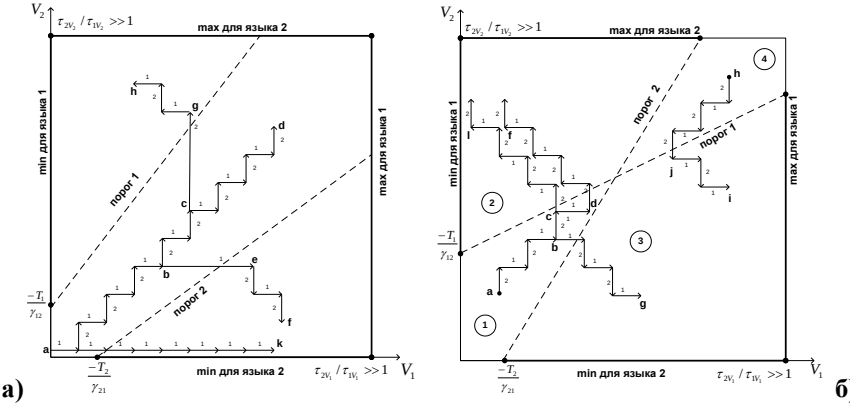


Рис. 3. Примеры движений изображающей точки (V_1, V_2) в случае T_1 и T_2 меньше нуля. Этапы обучения первому языку (горизонтальные линии) означены цифрой 1, а обучения второму языку (вертикальные линии) цифрой 2:

а) Траектория **abcd** соответствует процессу обучения и первому и второму языку. Траектория **ak** показывает процесс обучения только первому языку. Траектория **cgh** демонстрирует переход от обучения двум языкам к освоению только второго и торможения первого. Траектория **bef** демонстрирует переход от обучения двум языкам к освоению только первого и торможения второго.

б) Выделены четыре области процессов: (1); (2); (3) и (4). Параметры системы подобраны таким образом, что в области (1) возможно обучение двум языкам – траектория **abcd** до точки **d**; в области (2) возможно обучение второму языку, а первый язык тормозится – траектории **ce** и **df**; в области (3) возможно обучение первому языку, а второй язык тормозится - траектории **bg** и **ji;** в области (4) оба языка тормозятся – траектория **hj** до точки **j.**

4. Выводы

Приведенная феноменологическая модель спонтанного обучения и формирования языков (инструментов) общения позволила выделить области параметров, соответствующих экспериментально зафиксированным режимам при обучении детей [1–13].

1. Показано, что при больших порогах включения механизмов обучения и осознанного восприятия потока внешних сенсорных сигналов возможно блокирование спонтанного обучения. Известно, что эти пороги возрастают при неблагоприятных эмоциональных состояниях, а также недостаточном энергетическом обеспечении субъекта.

2. Показано, что спонтанное, естественное обучение нескольким (двум) языкам общения приводит к снижению объема каждого языка по сравнению со случаем обучения только одному языку. Причина этого заключается в трате выделенного на обучение временного ресурса на несколько языков,

вместо использования этого ресурса на освоение одного языка, предпочтительно родного.

3. Рассмотрены условия, при которых происходят переключения от освоения двух языков к обучению только одному языку и торможению другого или торможению обучения обоих языков.

Более подробное исследование возможных режимов функционирования и переключений между ними, несомненно, может быть проведено, если появятся специалисты (например, практические психологи), которым будет интересно интерпретировать регистрируемые ими экспериментальные факты.

Список литературы

1. Hoff E., Core C. Input and Language Development in Bilingually Developing Children // Semin Speech Lang. 2013; 34: 215—226. doi: http://dx.doi.org/10.1055/s-0033-1353448
2. Александрова Н.Ш. Может ли естественный билингвизм быть вреден? // Вестник РУДН. Серия: Вопросы образования: языки и специальность. 2017. Т. 14, № 2. С. 211–216.
3. Bialystok E., Craik F.I.M., Luk G. Cognitive control and lexical access in younger and older bilinguals // Journal of Experimental Psychology: Learning, Memory, and Cognition. 2008; 34: 859-873. URL: https://doi.org/10.1037/a0015638
4. Białystok E Bilingualism: The good, the bad, and the indifferent. // Bilingualism: Language and Cognition 12 (1), 2009, P. 3–11. doi: 10.1017/S13667289080034773
5. Bialystok E., Luk G., Peets K.F., Yang S. Receptive vocabulary differences in monolingual and bilingual children // Biling (Camb Engl). 2010 Oct; 13(4): P. 525–531. doi: 10.1017/S1366728909990423
6. Portocarrero J.S., Burright R.G., Donovick P.J. Vocabulary and verbal fluency of bilingual and monolingual college students.// Archives of Clinical Neuropsychology. 2007; 22: P. 415–422. https://doi.org/10.1016/j.acn.2007.01.015.
7. Александрова Н.Ш. Двуязычная среда - причина естественного билингвизма или лишь необходимое условие? // От билингвизма к транслингвизму: про и контра. Материалы III Международной научно-практической конференции. Москва, РУДН, 1–2 декабря 2017 г. 142-148.
8. Александрова Н.Ш. Почему Лера не заговорила по-немецки? К вопросу организации двуязычных детских садов / Curriculare und soziale Aspekte der Bildung und Erziehung bilingualer Kinder. Band 6; Berlin 2015; 273–275.
9. Александрова Н.Ш. Родной язык, иностранный язык и языковые феномены, у которых нет названия // Вопросы языкознания. 2006. № 3. С. 88-100.
10. Александрова Н.Ш. Билингвизм – адаптация в языковой сфере? //VI Международная конференция по когнитивной науке : тезисы докладов, Калининград, 23–27 июня 2014 г. С. 114–115.
11. Köpke B. Language attrition at the crossroads of brain, mind, and society https://hal.archives-ouvertes.fr/hal-00981119/document Submitted on 20 Apr 2014.
12. Schmid M. & Lowie W. editors. Modeling Bilingualism: From Structure to Chaos. In Honor of Kees de Bot. 308 pp John Benjamins. 2011.

13. Александрова Н.Ш. О чем говорят единичные наблюдения естественного билингвизма? Может ли помочь математика? // Казань, 2017.
14. Рабинович М.И., Трубецков Д.И. Введение в теорию колебаний и волн // 1984. Москва : Наука. 423 с.
15. Нуйдель И.В., Яхно В.Г. Моделирование процессов преобразования сенсорной информации // Нейрокомпютеры как основа мыслящей ЭВМ. 1993. С.207–224.
16. Яхно В.Г. Модели нейроноподобных систем. Динамические режимы преобразования информации // Материалы школы «Нелинейные волны 2002». Нижний Новгород: ИПФ РАН. 2003. С. 90-114.
17. Яхно В.Г., Макаренко Н.Г. Поможет ли нам создание «Цифрового двойника человека» лучше понимать друг друга? // Подходы к моделированию мышления, глава 6, с.169–202 / под ред. В. Г. Редько. Москва : ЛЕНАНД. 2014. – 392 с. (Синергетика от прошлого к будущему. № 70, Науки об искусственном. № 13).

НАУЧНЫЕ ТРУДЫ МОЛОДЫХ СПЕЦИАЛИСТОВ

В.И. ТЕРЕХОВ, А.А. САБИРОВ, И.М. ЧЁРНЕНЬКИЙ, В.М. ЧЁРНЕНЬКИЙ
Московский государственный технический университет им. Н.Э. Баумана
terekchow@bmstu.ru, sabirovdev@gmail.com

ПРЕДОБРАБОТКА SAR-ИЗОБРАЖЕНИЙ ДЛЯ АНАЛИЗА ЛЕДОВОЙ ОБСТАНОВКИ МЕТОДАМИ ГЛУБОКОГО ОБУЧЕНИЯ

В работе исследуется эффективность специализированных алгоритмов предобработки изображений, полученных радаром с синтезированной апертурой (SAR) с целью улучшения их качества. Изображения, предобработанные методами глубокого обучения, используются для распознавания границ ледового покрова, айсбергов и разреженных льдин. Результаты, полученные при обучении глубокой нейронной сети, показывают, что предварительная обработка данных SAR изображений играет определяющую роль в анализе ледовой обстановки.

Ключевые слова: радар с синтезированной апертурой, качество изображения, предобработка данных, глубокое обучение, нейронная сеть.

Введение

Дрейфующий лед может неожиданно блокировать каналы движения судов или корабли не соответствующего ледового класса, поэтому дешифрирование спутниковых изображений морских льдов, полученным радаром с синтезированной апертурой (SAR), используется для создания карт-схем ледовой обстановки, предотвращающих негативные ситуации. В настоящее время при анализе полученных изображений, эксперт классифицирует типы льда и делает карты-схемы ледовой обстановки практически в ручном режиме, что является долгой, сложной и дорогостоящей операцией.

Автоматизацию процесса анализа ледовой обстановки может значительно ускорить запуск в 2020-2021 годах спутников CubeSat [1] – инновационной и экономичной орбитальной группировки малых космических аппаратов (МКА), которые при низкой стоимости и простоте изготовления обеспечивают улучшенную автономную пространственную и временную разрешающую способность. При этом технология подготовки распознавания изображений с CubeSat и действующих спутников Sentinel, оснащенных

SAR, отличаться не будет, преимущество группировки CubeSat будет заключаться в скорости доставки и меньшем времени обработки сцены интереса (1–6 часов) и увеличению возможностей при анализе ледовой обстановки.

На сегодняшний день существует множество методов распознавания изображений обладающих различными особенностями [2, 3], однако авторы полагают, что применение метода глубокого обучения способно полностью автоматизировать процесс распознавания границ ледового покрова, айсбергов и разреженных льдин для определения районов, пригодных для прохождения судов соответствующих классов, на основе оперативных радарных изображений. В такой постановке задача, несомненно, является актуальной.

Как и все радары, SAR полагается на связь между излучаемым и отраженным радиосигналом для обнаружения различных свойств исследуемой области. При этом радарные сигналы нуждаются в предварительной обработке для учета геометрических искажений (например, перехода и ракурса) и различий в условиях освещения из-за топографии и освещаемой поверхности с учётом разных ракурсов съёмки целевой поверхности со спутника. Кроме этого, дополнительный шаг (Speckle Noise) необходим для удаления шума, вызванного отражениями от объектов, которые не представляют интереса. Само удаление осуществляется процессом, называемым Speckle-Filtering. Одними из наиболее часто используемых являются адаптивные фильтры (Frost, Lee), которые, используя локальную статистику для фильтрации данных, уменьшают помехи изображения, а в некоторых случаях сохраняют или улучшают края и другие функции [4–6].

Типичная последовательность обработки, применяемая к SAR-изображениям, включает в себя многоцелевой просмотр, speckle–filtering, ортотрансформирование и радиометрическую калибровку, коррекцию освещенности местности и мозаику – агрегацию изображений меньшего размера в единую панораму с применением сглаживания участков стыка [7, 8].

Как и в случае оптических данных, к данным SAR применяют ортотрансформацию и радиометрическую калибровку. Такая предобработка устраняет эффекты перспективы изображения (наклона) и некорректного отображения рельефа с целью создания контурно-правильного изображения. Результирующее ортотрансформированное изображение имеет постоянный масштаб, в котором элементы представлены в «истинных» положениях, позволяя точно измерять расстояния, углы и площади объектов. Процесс позволяет получить изображения, пригодные для сравнения/сопоставления экземпляров набора данных в случае, если экземпляры были получены при разных условиях снимка.

Для коррекции геометрических и радиометрических искажений, присутствующих в изображениях, собранных по крутому ландшафту, применяется дополнительный этап коррекции освещенности местности. Эти искажения маскируют полезное обратное рассеяние, связанное с земным покровом или

геофизическими особенностями, и их необходимо корректировать для эффективного картирования и мониторинга земного покрова с использованием данных SAR.

Исправленные данные SAR, полученные по разным спутниковым трекам, могут быть обработаны алгоритмами построения мозаики (панорамы) [7] для обеспечения охвата всего района. На сегодняшний день доступны как автоматические, так и ручные методы для работы с перекрывающимися областями изображений, что позволяет создать готовую к анализу бесшовную мозаику.

В [9, 10] показано, что качество входного изображения напрямую влияет на результат точности обучения глубокой нейросети. Однако исследования проводились на цветных экземплярах с целевыми объектами, составляющими от 30% до 90% объёма изображения и высокой контрастностью между объектом и фоном, что упрощает задачу сегментации. Снимки SAR значительно отличаются уровнем шума, более пологим переходом градиента на границах распознаваемых объектов и их однородностью, что усложняет задачу сегментации. Таким образом, одной из основных проблем является поиск наилучшего процесса обработки изображений для максимизации показателей обучения сети.

Последовательность применения алгоритмов

В данной статье рассматривается следующая последовательность применения алгоритмов обогащения, очищения и коррекции SAR-изображений (рис. 1).

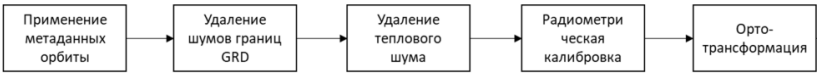


Рис. 1. Последовательность применения алгоритмов обработки SAR-изображений

1. Применение файла метаданных орбиты: файл содержит восстановленные векторы состояния орбиты – OSV [6] (векторы в декартовой системе координат или фиксированной системе координат Земли).
2. Удаление шумов границ GRD (Ground Range Detected) [4]: шумов низкой интенсивности и неверные данные на краях сцены.
3. Удаление теплового шума: аддитивный шум в суб-полосах, чтобы помочь уменьшить разрывы между суб-полосами для сцен в режимах захвата с несколькими полосами.
4. Радиометрическая калибровка: интенсивность обратного рассеяния, используя параметры калибровки датчика в метаданных GRD.

5. Ортотрансформация: преобразование данных из геометрии наземного радиуса действия, которое не учитывает рельеф местности, с использованием матрицы высот (для высоких широт).

Так, например, затруднительно найти отличительные области в изображениях, представленных на рис. 2. Однако использование первого снимка (рис. 2а) для анализа, учитывающего координаты и размерности объектов с большой вероятностью приведёт к ложным результатам. На изображении после ортотрансформации (рис. 2б) объекты местности уже имеют «реальные» координаты и размерности. Для применения алгоритма ортотрансформации необходимо построить цифровую модель рельефа (DEM – Digital Elevation Model).

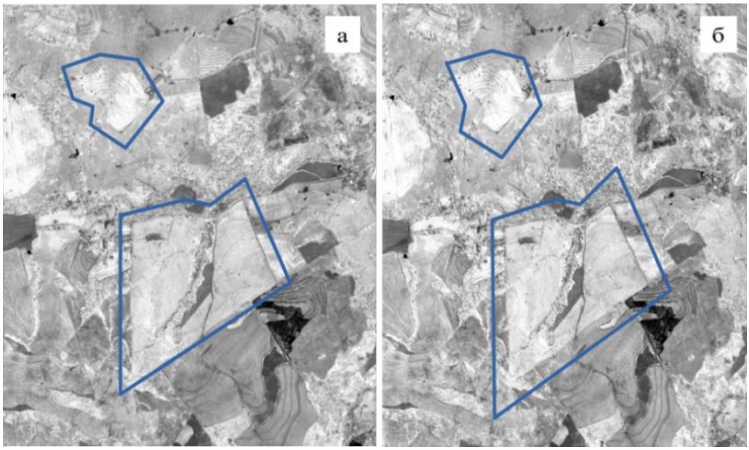


Рис. 2. Результат применения ортотрансформации (а) – исходное изображение, (б) – изображение после обработки (крупные отличительные объекты выделены контуром)

Основные типы информационных продуктов

На сегодняшний день существует два типа основных общепринятых информационных продукта (результат обработки сырых SAR-данных) для Sentinel – 1 (спутниковая группировка радиолокационной спутниковой системы Европейского космического агентства, состоящая из двух полярно-орбитальных спутников, работающая днем и ночью и получающая изображения в любое время независимо от погоды) [4, 6]:

1. GRD – имеет приблизительно квадратное пространственное разрешение и квадратное расстояние между пикселями с незначительной спекл-фильтрацией благодаря применённому Multi-looking [12] алгоритму. Набор данных

содержит векторы шума для последующего опционального применения алгоритмов коррекции по этим данным.

2. SLC – включает в себя один вид в каждом измерении, используя полную доступную пропускную способность сигнала с сохранением информации о фазе [13, 14].

Каждый из информационных продуктов подлежит дополнительным процедурам обработки, которые зависят от типа продукта, области применения (например, GRD уже отсутствует информация о фазе) и местности – в зависимости от распознаваемых объектов конкретные реализации алгоритмов, направленных на подавление шума, могут ошибочно принимать ключевые признаки объекта за шум, тем самым снижая общее качество входных данных для нейросети.

Оценка качества предобработанного SAR-изображения

За основу оценки качества обработки SAR-снимка был принят экспертный подход на основе голосования, т.н. процедура Борда [15], состоящий из следующей последовательности шагов.

1. Каждый эксперт строго ранжирует все альтернативные предобработанные SAR-изображения по их качеству.

2. Лицо, принимающее решение (ЛПР) подсчитывает число индивидуальных строгих упорядочений одного и того же вида.

3. Для каждого изображения ЛПР подсчитывает общее число экспертов, предпочитающих данное изображение всем остальным изображениям при парных сравнениях во всех полученных упорядочениях.

4. Для каждого изображения ЛПР вычисляет значение модифицированного критерия Борда: $Y_j = \sum_{k=1}^{m}(z_{j,k} - z_{ki})$, где $z_{j,k}$ – число экспертов, которые предпочитают качество изображения j качеству изображения k при их парном сравнении.

5. ЛПР упорядочивает все альтернативные изображения по убыванию значения модифицированного критерия Борда. Наилучшим согласно выражению $Y_l = \max_j Y_j$ считают изображение l, которое имеет максимальное значение по модифицированному критерию Борда.

Результаты

Результат предобработки SAR-снимка и соответственно его качество на выходе напрямую влияет на способность нейросети извлекать признаки изображения, что является одним из главных составляющих общей точности сети [9]. Чем выше способность сети извлекать признаки – тем выше шанс идентификации более «мелких» и «сглаженных» компонентов сложного целевого объекта.

Из общего набора предобработанных экземпляров были выбраны наилучшие результаты для каждого из типов. На рис. 3 представлены результаты

предобработки SAR-изображений относительно двух основ результатов обработки данных SLC и GRD.

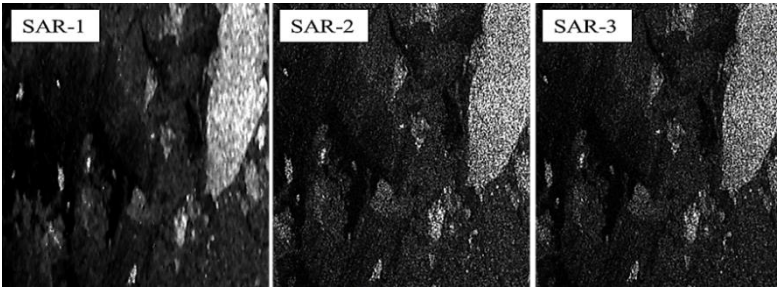


Рис. 3. Результат предобработки SAR изображений. SAR-1 – конвертация SLC информационного продукта в GRD, SAR-2 – GRD информационный продукт с повторным применением калибровки, SAR-3 – исходный GRD продукт

- SAR-1 – результат конвертации SLC информационного продукта в GRD в соответствии с порядком – удаление теплового шума, калибровка, Multi-looking, спекл-фильтрация (реализация Lee Sigma [16]), SRGR конвертация (по принципу линейной интерполяции).
- SAR-2 – GRD информационный продукт с повторным применением калибровки.
- SAR-3 – исходный GRD информационный продукт.

Согласно принятому методу оценки качества предобработанного SAR-изображения по процедуре Борда ниже представлена результирующая таблица (табл. 1).

Таблица 1

Результаты проведения процедуры Борда

Вариант	SAR-1	SAR-2	SAR-3
SAR-1	-	4	3
SAR-2	3	-	5
SAR-3	4	3	-

По результатам процедуры Борда был выбран экземпляр SAR-2.

Для каждого из рассматриваемых экземпляров был подготовлен аналогичный набор данных размеченный полигонами, характеризующих площадь разреженных льдин и ледового покрова.

Исходя из задачи – в качестве архитектуры была выбрана сверточная сеть [17], состоящая только из сверточных слоев без каких-либо полностью связанных слоев (fully convolutional network, FCN) [18]. На вход сети подавались предобработанные экземпляры в формате 8-bit grayscale размером

512×512 px, которые представляли минимальные области интереса из исходной мозаики (25316×19327 px).

Для сравнения результатов по каждому набору данных наибольшее внимание акцентировалось на задачу сегментации. Архитектура модели, методы аугментации и аппаратные средства не подлежали изменениям среди всех итераций по наборам. Результаты процесса обучения модели для наборов данных SAR-1, SAR-2, SAR-3 приведены в табл. 2.

Таблица 2

Результаты процесса обучения модели

Набор данных	Точность на этапе валидации	Значение функции потерь	Кол-во эпох (1000 шагов каждая)
SAR-1	0,89	0,208	417
SAR-2	0,94	0,101	401
SAR-3	0,91	0,199	425

Результаты сегментации для наборов SAR-1, SAR-2, SAR-3 приведены на рис. 7. Каждый результат состоит из трех частей: исходное изображение, маска областей (вывод нейросети), произведение первых двух матриц. На результирующей третьей части вручную отмечены области ошибок вывода нейросети в сравнении с размеченным исходным экземпляром. Для оценки инференции сети использовалось изображение SAR-2.

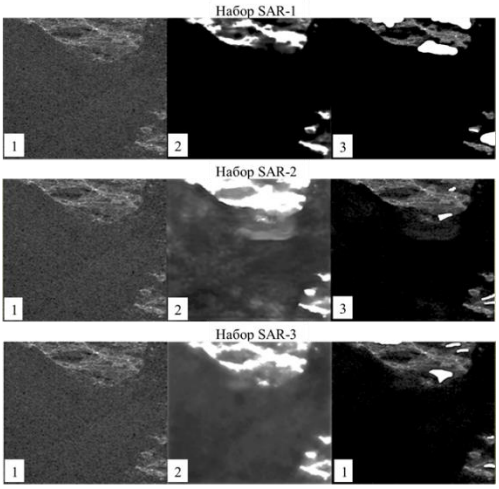


Рис. 7. Результаты сегментации для наборов SAR-1 − SAR-3.
1 − исходное изображение, 2 − маска-результат инференции сети,
3 − произведение изображения на маску и области ошибок

Заключение

Несмотря на низкое качество выборки SAR-1, сеть архитектуры FCN смогла преодолеть порог точности 0,8, после чего наблюдалось переобучение, независимо от частоты аугментации. Выборка SAR-2 информационного продукта GRD, имеющего отличительную видимую резкость, позволила преодолеть порог 0,9. Максимальные результаты были достигнуты после повторной калибровки продукта GRD, что привело к большей резкости перехода градиента границ целевого объекта. Вычислительные операции проводились при использовании точности fp32 (single-precision floating point). Хотя каждый слой данной архитектуры сети полностью поддерживает вычисления на основе fp16 (half-precision), применение данного подхода позволило увеличить скорость матричных вычислений и объём входной партии матриц на GPU, порог переобучения значительно снизился (SAR-1 – 0,65, SAR-2 – 0,69, SAR-3 – 0,73).

Как показывает практика, набор последовательных алгоритмов предобработки SAR-изображений является тем минимумом, который позволяет продолжить дальнейшую работу в части подготовки набора данных и проведения этапов тренировки нейросети. Подготовка «сырых» SAR-изображений на основе информационных продуктов типа SLC позволяет проводить более гибкую настройку предобработки данных, однако целесообразность подбора наилучших реализаций алгоритмов и их последовательности сомнительна при условии возможности использования продуктов типа GRD, оснащённых уже необходимыми метаданными (векторы теплового шума), для последующих экспериментов. В ряде случаев (например, при работе с источниками, имеющими низкую разрешающую способность) преодолеть целевой порог точности сети не представляется возможным, в силу большой зашумленности изображения, так что веса для пикселей, представляющих в действительности полезную нагрузку, могут соответствовать участку интенсивного шума. Однако чрезмерное использование алгоритмов удаления шума может привести к «размытым» и неявным границам объектов (как показал эксперимент SAR-1), в особенности для поверхностей водного пространства, что приводит к ухудшению показателей точности нейросети. Информационный продукт GRD (одним из способов улучшения качества которого является использование радиометрической калибровки, что подтверждено экспериментом SAR-2) является наилучшим продуктом. Также необходимо учитывать вычислительные аппаратные мощности при выборе конкретной реализации алгоритма для каждого звена последовательности, т.к. в общем случае, чем точнее алгоритм предобработки, тем больше процессорного времени (CPU/GPU) займёт его выполнение. Наиболее часто в эту категорию попадают итеративные алгоритмы оптимизации, такие как градиентный спуск, которые применяются и в методах предобработки SAR. Хотя алгоритмы такого рода и претерпели ряд оптимизаций за счёт матричных вычислений на

GPU, однако асимптотически N итераций не могут быть сравнимы с пренебрегаемыми константными коэффициентами в альтернативных не итеративных решениях.

Направление дальнейших исследований заключается в модификации архитектуры нейросети для классификации большого количества параметров ледовых образований: формы, объёма, возраста, вида, распределения и т.д. Рост набора классов требует совершенствования автоматизации формирования набора данных для предотвращения переобучения нейронной сети.

Список литературы

1. Mission Design Division, Ames Research Center , Moffett Field: Small Spacecraft Technology State of the Art – NASA // IEEE Photovoltaic Specialists Conference (PVSC). 2015.
2. Князев Б. А., Черненький В. М. Слаборазреженные фильтры в распознавании изображений //Информационно-измерительные и управляющие системы. 2017. Т. 15, №. 2. С. 49-56.
3. Князев Б. А., Черненький В. М. Методика и модель кластеризации паттернов двигательной активности лица как преобразований метаграфов // Вестник Московского государственного технического университета им. НЭ Баумана. Серия «Приборостроение». 2014. № 4 (97).
4. Henri Maitre, John Wiley & Sons, Processing of Synthetic Aperture Radar (SAR) Images, 2010.
5. Lopes A., Touzi R., Nezry E. Adaptive speckle filters and scene heterogeneity // IEEE transactions on Geoscience and Remote Sensing. 1990. V. 28, N 6. P. 992-1000.
6. Chris Oliver, Shaun Quegan. Understanding Synthetic Aperture Radar Images // Sci. Tech. Publishing, 2004.
7. Fahnestock M. [et al.]. Digital sar mosaic and elevation map of the greenland ice sheet // Boulder, CO: National Snow and Ice Data Center. (CD-ROM). 1997.
8. Freeman A. Radiometric calibration of SAR image data // International archives of photogrammetry and remote sensing. 1993. V. 29. P. 212-212.
9. Dodge S., Karam L. Understanding how image quality affects deep neural networks // 2016 eighth international conference on quality of multimedia experience (QoMEX). – IEEE, 2016. – P. 1-6.
10. Koziarski M., Cyganek B. Impact of low resolution on image recognition with deep neural networks: An experimental study //International Journal of Applied Mathematics and Computer Science. – 2018. – V. 28. – N 4.
11. Fernández C. Sentinels POD service file format specifications // Eur. Space Agency, Paris, France, Tech. Rep. GMESGSEG-EOPG-FS-10-0075. 2011.
12. Schmitt A. Multiscale and multidirectional multilooking for SAR image enhancement // IEEE Transactions on Geoscience and Remote Sensing. 2016. V. 54, N 9. P. 5117-5134.
13. Abdurahman Bayanudin A., Heru Jatmiko R. Orthorectification of Sentinel-1 SAR (Synthetic Aperture Radar) Data in Some Parts of South-eastern Sulawesi Using Sentinel-1 Toolbox // IOP Conference Series: Earth and Environmental Science. 2016. V. 47. N. 1. P. 012007.

14. Huang G. M. [et al.]. Algorithms and experiment on SAR image orthorectification based on polynomial rectification and height displacement correction // International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 2004. – V. 34. N Part XXX.
15. Постников В. М., Черненький В. М. Методы принятия решений в системах организационного управления // Москва : Изд-во МГТУ им. НЭ Баумана. 2014.
16. Lee J. S. A simple speckle smoothing algorithm for synthetic aperture radar images // IEEE Transactions on Systems, Man, and Cybernetics. 1983. N 1. P. 85-89.
17. Ian Goodfellow, Yoshua Bengio, Aaron Courville: Deep Learning // MIT Press. – 2016.
18. Ronneberger O., Fischer P., Brox T. U-net: Convolutional networks for biomedical image segmentation // International Conference on Medical image computing and computer-assisted intervention. – Springer, Cham, 2015. – P. 234-241.

И. М. ГАДЖИЕВ, С. А. ДОЛЕНКО
Московский государственный университет им. М.В. Ломоносова
ismail.gadzhiev@yahoo.com

МЕТОДЫ ОБЪЕДИНЕНИЯ КЛАССОВ В ПРОЦЕССЕ РАБОТЫ АЛГОРИТМА АДАПТИВНОГО ПОСТРОЕНИЯ ИЕРАРХИЧЕСКИХ НЕЙРОСЕТЕВЫХ КЛАССИФИКАТОРОВ

Объектом исследования настоящей работы является иерархический нейросетевой классификатор (ИНК), использующий специальный алгоритм для решения задачи многоклассовой классификации. В процессе работы каждого узла ИНК схожие классы объединяются в группы; последующее разделение групп осуществляется рекурсивным применением алгоритма на следующих уровнях иерархии. В данной работе рассматривается проблема случайного слияния классов и сравниваются между собой различные алгоритмы объединения классов в группы.

Ключевые слова: нейронные сети, деревья решений, иерархический нейросетевой классификатор, слияние классов.

Введение

Алгоритм адаптивного построения иерархических нейросетевых классификаторов сочетает в себе некоторые сильные стороны нейронных сетей и деревьев решений. В случае если в данных существует некоторая иерархия классов, он должен уменьшать смещение предсказаний модели.

В процессе работы алгоритма возникает проблема случайного слияния классов, которая решается введением порога активаций, что было отмечено в предыдущих работах, посвящённых алгоритму [1, 2]. Однако подбор порога активаций может быть трудоёмкой задачей, поэтому необходимо разработать методы его адаптивной установки. Успешное решение проблемы случайного слияния классов позволит применять алгоритм для решения сложных задач, например, задач распознавания изображений.

Нейросетевые деревья

В литературе ранее уже предлагались способы построения нейросетевых деревьев. В работе [3] была предложена архитектура SAINT, схема которой изображена на рис. 1, в основе которой лежат нейронные сети Кохонена [4].

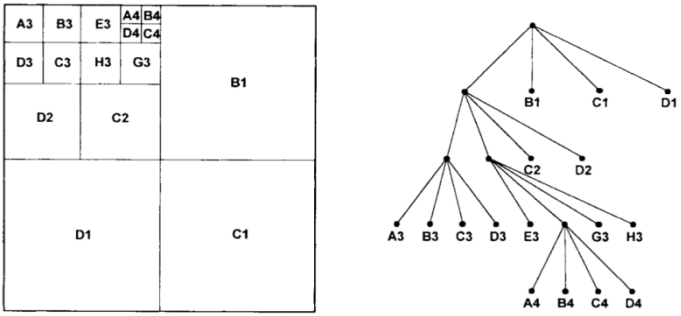


Рис. 1. Схема архитектуры SAINT

В работе предлагается использовать нейронную сеть Кохонена на исходных данных. После обучения необходимо удалить «мёртвые» нейроны, которые не представляют ни один из кластеров, затем объединить нейроны, отвечающие близким кластерам. В каждом полученном узле обучается новая сеть Кохонена. После построения иерархической структуры каждому конечному узлу присваивается метка класса, наиболее представленного в нём.

В работе [5] было представлено нейронное дерево с кластеризацией. Структура дерева выстраивается адаптивно. В узел помещается нейронная сеть; если качество ответов получается удовлетворительным, узел объявляется конечным. Иначе в узле происходит кластеризация, а каждому из получившихся кластеров сопоставляется новый узел. В качестве алгоритма кластеризации предлагается использовать K-Means (метод K-средних).

Иерархический нейросетевой классификатор

Иерархический нейросетевой классификатор (ИНК), алгоритм обучения которого описан в статье [1], предназначен для решения задачи многоклассовой классификации. Пусть обучающая выборка T состоит из примеров $\{X_i, y_i\}$, где y_i – правильный ответ для X_i, принадлежащий множеству классов G. Иерархический нейросетевой классификатор – это дерево, в каждом узле которого находится нейронная сеть с одним скрытым слоем, обучающаяся на некотором подмножестве всех данных с модифицированными ответами, т.е. обучающая выборка для узла n состоит из примеров $T_n = \{X_i, \hat{y}\}_n$, таких что $y_i \in G_n \subseteq G$, $\hat{y}_i$ – модифицированный ответ для X_i, $\hat{y}_i \in \widehat{G_n}$, где $\widehat{G_n}$ – множество меньшей мощности, полученное отображением множества G_n.

Дерево строится путём последовательного слияния классов, которые сеть чаще всего путает между собой. Опишем построение дерева в узле n. Сеть с одним скрытым слоем обучается на исходных данных $T_n \subset T$. После определённого заранее количества эпох N_{start} производится голосование на примерах из T_n. Если большинство примеров из класса j были определены сетью как примеры, принадлежащие классу i, то эти два класса сливаются в один, а

желаемым ответом для всех примеров класса j становится класс i, после чего обучение сети продолжается. Каждый раз после $N_{timeout}$ эпох после предыдущего объединения голосование производится вновь. В качестве критерия остановки используется остановка по валидационному набору. Таким образом, после остановки обучения сеть выделяет из множества классов G_n несколько (K) подмножеств, элементы каждого из которых отображаются в один и тот же элемент $\widehat{G_n}$. Мы получаем для узла n K дочерних узлов, каждый из которых затем тренируем по описанному выше алгоритму.

Проблема случайного слияния классов

При голосовании нейронов на примерах из обучающей выборки существует вероятность случайного слияния классов. Это может произойти, когда сеть сильно недообучена и большое влияние на ответ имеет начальное распределение весов.

Для иллюстрации рассмотрим задачу **vowels** – задачу распознавания английских гласных [6]. Тренировочный набор состоит из 660 примеров, количество признаков – 10, количество классов (различных гласных) – 11. Результаты оценивались на независимой выборке (тестовом наборе) из 110 примеров.

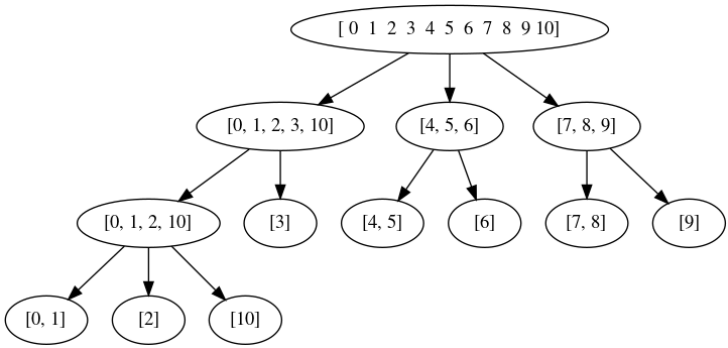


Рис. 2. Дерево ИНК, построенное на задаче **vowels**

На рис. 2 изображено дерево, построенное иерархическим нейросетевым классификатором. Доля правильных ответов при классификации на независимой выборке составила 49 процентов. Низкая доля правильных ответов обусловлена нежелательными слияниями классов.

Для того чтобы избежать случайного слияния классов, можно ввести порог активаций θ. На этапе голосования учитываются голоса только тех примеров, для которых максимальная активация выходного нейрона больше порога, т.е. $f_j(X_i) > \theta$, где j – номер нейрона. На рис. 3 изображена зависимость средней доли правильных ответов при классификации и её стандартное отклонение на валидационном наборе от выбора порога.

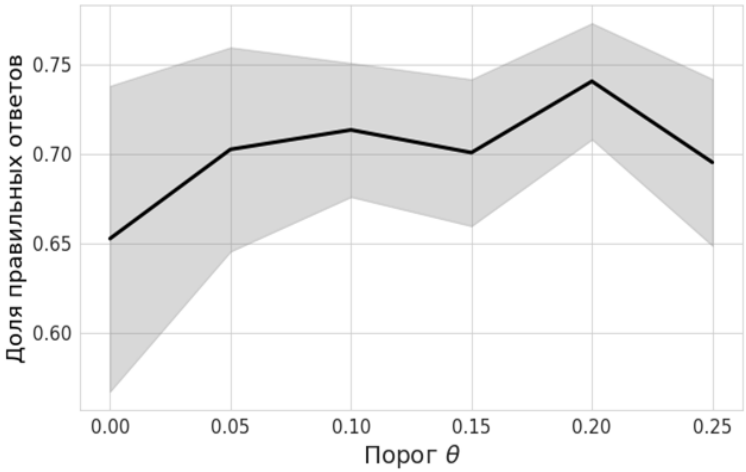


Рис. 3. Зависимость доли правильных ответов от порога

Алгоритм обучался на фиксированном тренировочном наборе с каждым порогом $N = 10$ раз, итоговая доля правильных ответов оценивалась на независимом наборе. В скрытом слое каждой из сетей содержатся два нейрона. Как видно, с увеличением порога доля правильных ответов увеличивается, а её разброс уменьшается. Однако затем доля правильных ответов падает, так как высокий порог запрещает любые слияния классов

Подбор порога можно осуществлять с помощью перебора по некоторому множеству возможных значений. Эта процедура может быть трудоёмкой; кроме того, порог выбирается сразу для всех узлов и является некоторым компромиссом. Более эффективным может являться выбор порога для каждого из узлов. Решение этой задачи перебором является вычислительно дорогим, что сводит на нет одно из преимуществ алгоритма.

Рассмотрим различные подходы к выбору значения порога активаций, по которому производится голосование примеров в узлах ИНК.

В данной работе использовалась программная реализация алгоритма адаптивного построения ИНК, созданная автором на языке Python 3.6 с использованием библиотек Keras [7] и Tensorflow [8].

Статистический подход к выбору порога

Задача состоит в локализации области, где сосредоточены значения активаций нейронов при случайной инициализации весов нейронной сети в каж-

дом из узлов. Чтобы оценить положение среднего и ширину области локализации, можно воспользоваться статистическими показателями ещё не обученной сети. Необходимо принять во внимание, что лучше всего оценивать именно безусловное среднее, которое не зависит от конкретных значений весов при инициализации.

Для этого можно несколько раз инициализировать модель в узле n, и для каждой инициализации вычислить среднее значение активации по всем нейронам выходного слоя в узле μ_{nl} и стандартное отклонение активации выходного слоя в узле σ_{nl}, где l =1, ...L – количество инициализаций. Затем использовать в качестве порога в узле n значение:

$$\theta_n = \frac{1}{L}\sum_{l=1}^{L}(\mu_{nl} + \sigma_{nl}). \quad (1)$$

В качестве активационной функции выходного слоя каждой из сетей используется softmax. Можно заметить, что при случайной инициализации весов в силу полносвязности нейронной сети ни один из выходов не выделен относительно другого, а сумма активаций всегда равняется единице. Таким образом, среднее значение активации при случайной инициализации для функции softmax всегда равняется $\frac{1}{N_o}$, где N_o – число выходных нейронов. Для оценки области локализации активаций при инициализации в узле n достаточно вычислить величину:

$$\theta_n = \frac{1}{N_n} + \frac{1}{L}\sum_{l=1}^{L}\sigma_{nl}. \quad (2)$$

Через N_n обозначено количество выходных нейронов для сети в узле n.

В качестве правой границы интервала можно также принять среднее значение, что снижает вычислительную стоимость метода:

$$\theta = \frac{1}{N_n}. \quad (3)$$

Выбор своего порога в каждом узле может привести к переобучению из-за слишком сильной подстройки к тренировочному набору. Для того чтобы избежать этого, можно выбрать общий компромиссный порог для всех узлов. Рассмотрим узел n, в котором распознаются примеры из N_n классов. Тогда в узле n можно установить значение порога согласно (3). Заметим, что в результате работы алгоритма могут появиться только узлы, для которых $N_n < K$, где K – исходное число классов. Не зная заранее вид дерева, предположим, что в дереве появятся узлы со всеми возможными значениями N_n, т.е. K, K–1, ..., 3. Тогда в качестве порога мы можем взять среднее значение от всех возможных значений порога по (3):

$$\theta = \frac{1}{K-2}\sum_{n=3}^{K} 1/n. \quad (4)$$

Суммирование активаций

Рассмотрим другие возможные подходы к решению проблемы случайного слияния классов.

Самый наивный подход заключается в учёте не целых голосов, а именно активаций выходных нейронов. Таким образом, можно увеличить влияние "уверенных" ответов нейронов.

Пусть $\widehat{f}_i(X_j)$ – активация i-го выходного нейрона на *j*-м примере. Пусть имеется K классов $G_k \subset \mathrm{G}$. Рассмотрим голосование примеров *m*-го класса $\mathrm{X}_\mathrm{j}^{(\mathrm{m})} = \{\mathrm{X_j}: \mathrm{y_j} \in \mathrm{G_m}\}$. Тогда для *i*-го класса

$$s_i^{(m)} = \sum_j \widehat{f}_i(X_j^{(m)}). \quad (5)$$

Соответственно, мы выбираем класс с наибольшим значением *s*.

Естественным продолжением идеи предыдущего пункта является суммирование активаций с весами w_i, которые выделяют среди всех голосов наиболее «уверенные»:

$$s_i^{(m)} = \sum_j w_i(X_j^{(m)})\widehat{f}_i(X_j^{(m)}). \quad (6)$$

Веса $w_i(X)$ мы можем подобрать, например, из следующих соображений: чем больше изменилась активация нейрона на данном примере за время обучения, тем больше он «обучился». Тогда обозначим $\widehat{f_i^{(0)}}(X)$ – активация *i*-го нейрона на примере *X* до начала обучения. Тогда

$$w_i(X) = |f_i(\mathrm{X}) - \widehat{f_i^{(0)}}(X)|. \quad (7)$$

Понятно, что в качестве меры отклонения от начальной активации можно было использовать любую другую монотонно возрастающую функцию, например, квадрат отклонения. В данной работе использовалось именно абсолютное отклонение в связи с его меньшей чувствительностью к выбросам.

Результаты

На рис. 4 показаны результаты тестирования методов подбора порога на задаче распознавания гласных.

Алгоритм 10 раз обучался с разной инициализацией весов на фиксированном тренировочном наборе, каждый раз доля правильных ответов оценивалась на независимом наборе данных.

Чертой внутри коробки изображено медианное значение доли правильных ответов, синяя коробка включает в себя 50 процентов точек, положение горизонтальной черты сверху – сумма третьего квартиля и полутора межквартильных расстояний, снизу – разность первого квартиля и полутора межквартильных расстояний. Серой точкой обозначен результат запуска для метода 2, не попавший в вышеозначенный интервал.

В качестве базовой модели для иерархического нейросетевого классификатора использовался персептрон с двумя нейронами в единственном скрытом слое с активационной функцией скрытого слоя ReLU. Для обучения персептрона использовался стохастический градиентный спуск с моментом 0.9 и скоростью обучения 0.1.

Цифрами на рис. 4 обозначены методы слияния классов: 1 – оценка порога голосования по среднему (3), 2 – по среднему и стандартному отклонению (2), 3 – компромиссный порог для всех узлов (4), 4 – взвешенная сумма активаций (6), 5 – сумма активаций (5).

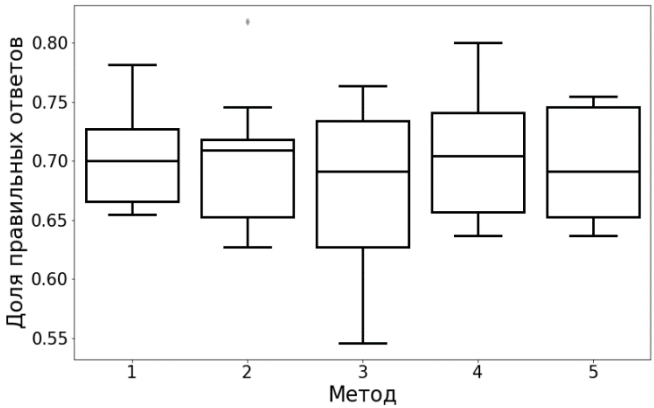


Рис. 4. Сравнение различных методов слияния классов. Пояснения даны в тексте

Исходя из полученных результатов, можно заключить, что оптимальным методом контроля слияния классов является оценка порога средним значением активации в узле (метод 1 на рис.4). Этот метод сочетает в себе сравнимые с остальными результаты и низкую вычислительную стоимость.

Впрочем, для того, чтобы провести более корректное сравнение методов, необходимо протестировать их на ряде других задач.

Выводы

В результате работы были получены и протестированы методы, позволяющие решать проблему случайного слияния классов в ходе работы алгоритма адаптивного построения иерархического нейросетевого классификатора: задание порога по безусловному среднему значению активации при инициализации нейронной сети, по безусловному среднему и стандартному отклонению, задание компромиссного порога, суммирование активаций с весами и без.

Показано, что метод контроля случайного слияния классов с помощью задания порога по среднему значению активации в узле является оптимальным с точки зрения вычислительной стоимости и сравнимым с остальными методами с точки зрения получаемых результатов классификации. В частности, для задачи распознавания гласных, рассмотренной в данной работе, рекомендуется использовать именно этот способ.

В целом все предложенные методы позволяют решить проблему случайного слияния классов и показывают близкие результаты, за исключением использования компромиссного порога для всех узлов.

Список литературы

1. Svetlov V.A., Persiantsev I.G., Shugay J.S., Dolenko S.A. A new implementation of the algorithm of adaptive construction of hierarchical neural network classifiers // Optical Memory and Neural Networks (Information Optics). 2015. V. 24, N 4. P. 288–294.
2. Svetlov V.A., Dolenko S.A. Development of the Algorithm of Adaptive Construction of Hierarchical Neural Network Classifiers // Optical Memory and Neural Networks (Information Optics). 2017. Vol. 26. No. 1. P. 40–46.
3. Song H.H., Lee S.W. A Self Organizing Neural Tree for Large Set Pattern Classification // IEEE Transactions on Neural Networks. 1998. Vol 9, No.3. P. 369–380.
4. Kohonen T. Self-organized formation of topologically correct feature maps // Biological Cybernetics. 1982. Vol. 43. No. 1. P. 59–69.
5. Zhong Qiu Zhao, De Shuang Huang, Lin Wei Guo. A Novel Clustering Neural Tree for Pattern Classication // IEEE International Joint Conference on Neural Networks. 2004. DOI: 10.1109/IJCNN.2004.1380132
6. Dua D., Graff C. (2019). UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science. Connectionist Bench (Vowel Recognition - Deterding Data) Data Set. https://archive.ics.uci.edu/ml/datasets/Connectionist+Bench+(Vowel+Recognition+-+Deterding+Data).
7. Keras: The Python Deep Learning library. https://keras.io.
8. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. http://tensorflow.org/.

М. А. ГУНЬКИН, В. И. ТЕРЕХОВ, В. Ю. ФОМИН
Московский государственный технический университет им. Н.Э. Баумана
misha36gunkin@amil.ru, terekchow@bmstu.ru, vova-fomin97@mail.ru

ПРЕДИКТИВНАЯ МОДЕЛЬ СПРОСА НА ТЕАТРАЛЬНЫЕ БИЛЕТЫ С ПРИМЕНЕНИЕМ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

В статье проводится анализ спроса и продаж театральных билетов с использованием различных методов машинного обучения. Проведенные исследования показали, что с помощью методов машинного обучения можно значительно улучшить прогноз продаж театральных билетов на спектакли. Результатом работы является исследование шести предиктивных моделей, из которых была выбрана модель на основе рекуррентной LSTM сети, дающая наименьшую ошибку прогноза. В заключении приведены направления дальнейших исследований.

Ключевые слова: анализ данных, прогнозирование, методы машинного обучения, нейронные сети, предиктивная модель.

Введение

В условиях существенного повышения качества жизни в крупных мегаполисах резко возрастает интерес населения к культурной жизни. В частности, возрастает количество зрителей, посещающих театры, выставки, музеи, концертные площадки, кинотеатры, городские массовые мероприятия и т.д.

В этих условиях, а также в условиях обостряющейся конкуренции, организации, основным бизнесом которых является сфера культуры, приходят к пониманию того, что качество проведения как самих мероприятий, так и в особенности политики цен на продаваемые билеты должны стать более продуманными и гибкими. С учетом того, что на сегодняшний день такая политика ведется на основе исторических данных и необоснованного назначения максимальной цены, решение задачи с применением методов машинного обучения, безусловно, является актуальным.

В статье приводится один из возможных подходов анализа спроса продаж театральных билетов с использованием различных методов машинного обучения в конкретной ситуации на данных, предоставленных одним из Московских театров.

Для исследования был получен набор данных о театральных представлениях, содержащий продажи билетов театром за некоторый промежуток времени. На предоставленном наборе данных, с применением методов машинного обучения, было проведено обучение нескольких моделей. Модель, которая достигла максимально возможного качества прогноза, авторы считают лучшей в решении поставленной задачи.

Постановка задачи

Провести анализ продаж билетов театра на основе данных о продажах за два года и построить предиктивную модель, на основе которой возможно прогнозировать продажи театра в конкретный день, на конкретный спектакль, с учетом цен на билеты. Требуется достигнуть средней абсолютной ошибки не более 5 непроданных билетов по каждой категории мест. Такая ошибка была установлена заказчиком проводимого исследования.

Анализ данных

Предоставленный набор данных о театральных представлениях содержит 6 атрибутов, которые описывают 12 017 объектов (табл. 1). Каждый объект представляет собой категорию мест, имеющих одинаковую стоимость. При этом имеется два типа признаков: вещественные и категориальные.

К вещественным признакам относятся: цена мест; выделенное количество мест под данную категорию; количество непроданных мест. Категориальные признаки включают в себя дату, время и название представления.

Таблица 1

Дата сет о театральных представлениях (фрагмент)

Цена	Количество	Дата	Время	Название	Не продано
100	30	12.07.2018	19:00	Название 1	1
900	6	17.01.2019	19:00	Название 2	0
700	13	29.06.2017	19:00	Название 3	13
…	…	…	…	…	…
2000	14	20.06.2017	19:00	Название 4	8
300	6	14.03.2017	19:00	Название N	0

Для обучаемой модели дата в формате «день.месяц.год» не информативна, так как на целевой признак влияет только день недели и месяц. Следовательно, можно заменить признак «Дата» на два новых признака: «День недели» и «Месяц», являющиеся более информативными при решении поставленной задачи. В табл. 2 представлен набор данных с учетом проведенных операций.

Анализ категориальных признаков [1] показал, что время начала представления практически всегда принимает значение «19:00», а в остальных случаях от него практически не отличается. Так как время начала практически не меняется, данный признак является неинформативным, ввиду того, что разница времени начала спектакля в 30 минут не должна никак сказываться в спросе на билеты, поэтому данный признак можно из набора данных исключить. Остальные категориальные признаки имеют широкий спектр принимаемых значений и являются более информативными. Также стоит отметить, что некоторые из представленных спектаклей имеют довольно мало примеров (данных), что в дальнейшем может вызвать трудности в процессе обучения. Попарные зависимости между вещественными признаками представлены на рис. 1.

Таблица 2

Набор данных с указанием «Месяца» и «Дня недели» (фрагмент)

Цена	Количество	Месяц	День недели	Время	Название	Не продано
100	30	Июль	Четверг	19:00	Название 1	1
900	6	Январь	Четверг	19:00	Название 2	0
700	13	Июнь	Среда	19:00	Название 3	13
…	…	…	…	…	…	
2000	14	Июнь	Вторник	19:00	Название 4	8
300	6	Март	Пятница	19:00	Название N	0

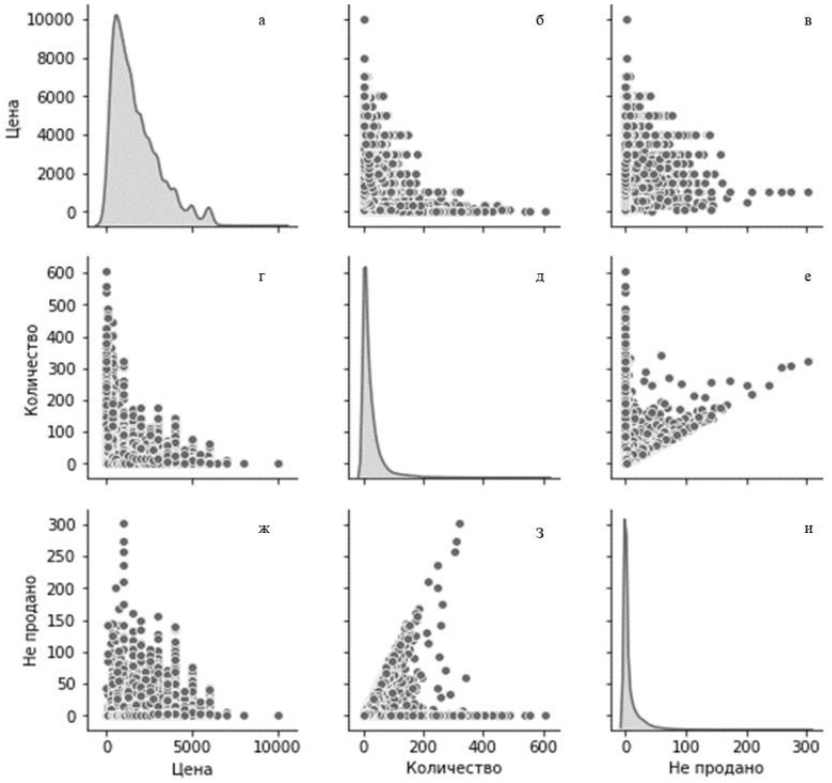


Рис. 1. Попарные зависимости между вещественными признаками

Анализ рис. 1 показывает, что, театр часто продает большое количество билетов по очень низкой цене (рис. 1г) и, следовательно, количество непроданных билетов стремится к нулю (рис. 1в). И наоборот, чем больше цена

билетов, тем больше их не продано. В остальных случаях зависимость предложенного количества билетов от непроданных билетов близка к линейной (рис. 1е).

В табл. 3 приведены рассчитанные коэффициенты корреляции между вещественными признаками.

Таблица 3

Таблица коэффициентов корреляции вещественных признаков

	Цена	Количество	Не продано
Цена	1,00	–0,13	0,23
Количество	–0,13	1,00	0,40
Не продано	0,23	0,40	1,00

Статистическая информация по вещественным признакам представлена в табл. 4.

Таблица 4

Статистическая информация

	Цена	Количество	Не продано
count	12342,00	12342,00	12342,00
mean	1418,43	20,58	5,74
std	1286,90	33,47	14,02
min	10,00	1,00	0,00
25%	500,00	4,00	0,00
50%	1000,00	9,00	0,00
75%	2000,00	24,00	5,00
max	10000,00	605,00	302,00

Признаки имеют разный масштаб, поэтому для качественного обучения моделей их необходимо нормализовать следующим образом:

$$x_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}},$$

где x_i – значение признака; $x_{\min}$– минимальное значение данного признака среди всех объектов; $x_{\max}$ – максимальное значение данного признака среди всех объектов.

Затем было выполнено one-hot [2] кодирование категориальных признаков. Такое кодирование заменяет все значения категориальных признаков на новый признак, который принимает значение 1 в случае соответствия данному объекту и 0 в противном случае. В результате проведенных операций данные стали описываться 41 признаком.

После анализа и обработки датасета данных театральных представлений можно перейти к этапу обучения моделей. Для обучения и проверки качества

датасет был разделен следующим образом: 70% – обучающая выборка, 20% - тестовая выборка, 10% – валидационная выборка.

Линейная регрессия

Попытка объяснить зависимость количества непроданных билетов от признаков с помощью линейной регрессии, как и ожидалось, не принесла хороших результатов. В модели, построенной на основе метода линейной регрессии средняя абсолютная ошибка на тестовой выборке получилась равной 8,8 непроданных билетов.

Случайный лес

Следующий метод машинного обучения – метод случайного леса [3]. Для подбора гиперпараметров [4] в этом случае использовался метод GridSearchCV [5]. В результате был построен случайный лес из 100 деревьев с ограничением на максимальную глубину деревьев, равную 40. В модели, построенной на основе метода случайного леса, средняя абсолютная ошибка на тестовой выборке сократилась до 5,8 непроданных билетов.

Градиентный бустинг

Гиперпараметры для градиентного бустинга [6] также выбирались с помощью GridSearchCV. В итоге модель имела следующие параметры: количество деревьев равно 200, максимальная глубина деревьев равна 5. В модели, построенной на основе метода градиентного бустинга, сильно сократить среднюю абсолютную ошибку на тестовой выборке не удалось, она оказалась равной 5,7 непроданных билетов.

Многослойный персептрон

В этом случае был выбран многослойный персептрон (MLP) с двумя скрытыми слоями [7]. В первом слое расположено 90 нейронов с функцией активации гиперболический тангенс. Во втором слое 60 нейронов с функцией активации RELU [7]. Выходной слой представлен одним RELU нейроном. Для борьбы с переобучением использовался dropout [8] с вероятностью исключения нейрона из сети, равной 0,2. Архитектура данной нейронной сети представлена на рис. 2.

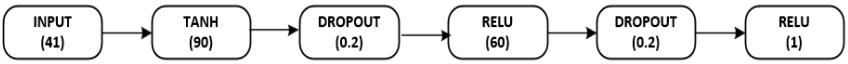


Рис. 2. Архитектура MLP

Зависимость средней квадратичной ошибки от эпохи представлена на рис. 3 а, а средней абсолютной ошибки от эпохи на рис. 3 б.

Средняя абсолютная ошибка модели, построенной с применением метода MLP, на тестовых данных принимает значение 3,4 непроданных билета.

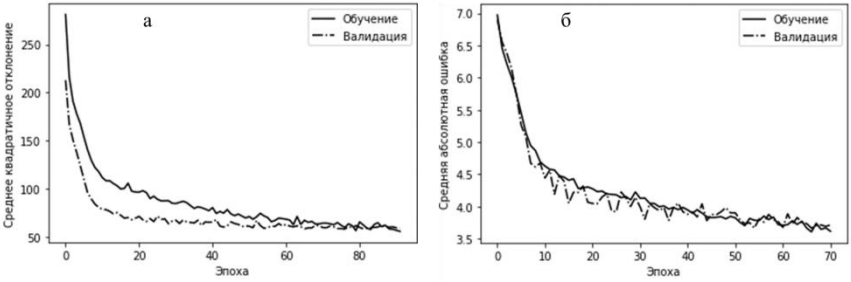


Рис. 3. Зависимости ошибки от эпохи для MLP: а – средней квадратичной ошибки, б – средней абсолютной ошибки

Радиально-базисная модель нейронной сети

В качестве функций активации радиально-базисная модель (RBF) нейронной сети [9] использует радиально-симметричные функции:

$$F(x) = e^{-\mathcal{B}(x-\mu)^2},$$

где μ – параметр, определяющий центр симметрии графика функции; $\mathcal{B}$ – параметр, отвечающий за разброс.

В сети используется два скрытых слоя. Первый слой линейной активации содержит 50 нейронов. Второй RBF-слой содержит 8 нейронов. Выходной RELU-слой состоит из одного нейрона. Для борьбы с переобучением была применена батч-нормализация [9].

Архитектура данной сети представлена на рис. 4.

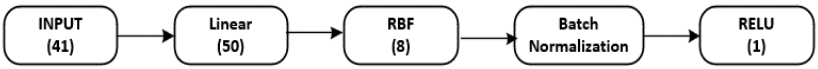


Рис. 4. Архитектура RBF нейронной сети

Зависимость средней квадратичной ошибки от эпохи представлена на рис. 5а, а средней абсолютной ошибки от эпохи, на рис. 5б.

Средняя абсолютная ошибка модели, построенной с применением метода RBF сети, на тестовых данных принимает значение 3,7 непроданных билета.

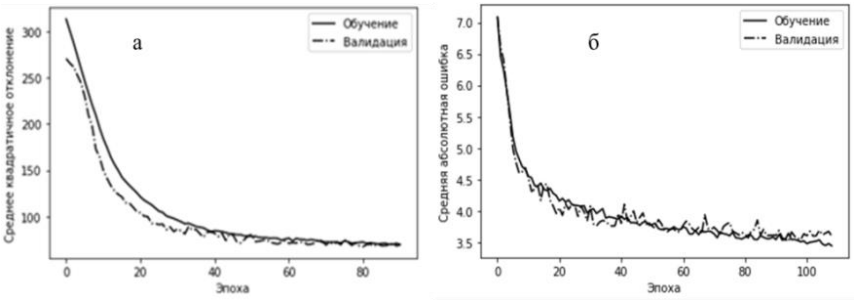


Рис. 5. Зависимости ошибки от эпохи для RBF нейронной сети: а - средней квадратичной ошибки, б - средней абсолютной ошибки

Рекуррентная LSTM сеть

В данном случае была применена одна из разновидностей архитектуры рекуррентных нейронных сетей – сеть долгой краткосрочной памяти (Long short-term memory; LSTM) [10, 11]. Такая сеть хорошо обучается на задачах обработки и прогнозирования временных рядов в случаях, когда важные события разделены временными промежутками с неопределённой продолжительностью и границами. Сеть содержит LSTM-модули представляющие собой рекуррентные модули, запоминающие значения на короткие и на длинные промежутки времени, так как не используют функцию активации.

В сети используется два скрытых слоя [12]. Первый LSTM-слой содержит 45 нейронов. Второй LSTM-слой содержит 30 нейронов. Выходной RELU слой состоит из одного нейрона. Для борьбы с переобучением использовался dropout с вероятностью исключения нейрона из сети равной 0,3.

Архитектура рекуррентной LSTM сети показана на рис. 6.

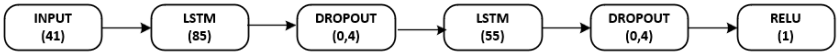


Рис. 6. Архитектура рекуррентной LSTM сети

График зависимости средней квадратичной ошибки от эпохи представлен на рис. 7.

Средняя абсолютная ошибка данной модели, построенной с применением метода рекуррентной LSTM сети, на тестовых данных принимает значение 3,3 непроданных билета.

Обобщение итоговых результатов прогнозирования всеми методами представлены в табл. 5.

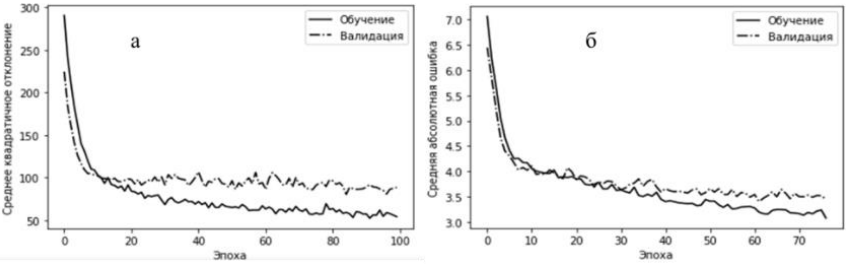


Рис. 7. Зависимости ошибки от эпохи рекуррентной LSTM сети: а – средней квадратичной ошибки, б – средней абсолютной ошибки

Таблица 5

Итоговые результаты прогнозирования

Модель на основе метода	Средняя абсолютная ошибка	Средняя квадратичная ошибка
Линейная регрессия	8,8	234,3
Случайный лес	5,8	180,7
Градиентный бустинг	5,7	168,5
Многослойный персептрон (MLP)	3,4	87,1
Радиально-базисная модель нейронной сети (RBF)	3,7	92,4
Рекуррентная LSTM сеть	3,3	76,4

Заключение

В проведенном исследовании были предложены и реализованы шесть способов прогнозирования спроса на билеты театра. По итогам анализа всех моделей, основанных на шести методах, можно сделать вывод, что лучшей предиктивной моделью по анализу спроса и продаж театральных билетов является модель, построенная на основе рекуррентной LSTM-сети, так как она имеет наименьшую среднюю абсолютную ошибку. Стоит отметить, что всегда стоит пробовать несколько моделей для определения наиболее подходящей для решения конкретной задачи. Безусловно, полученные результаты требуют проведения дальнейших исследований. Будет продолжена работа над улучшением качества анализа и прогнозирования, которое зависит, в первую очередь, от гиперпараметров представленных моделей, а также исследование новых моделей и решения задач в других предметных областях.

Список литературы

1. Флах П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. Москва : ДМК Пресс, 2015. 315 с.
2. Youn Y.S., Nam K.M., Song H.J., Kim J.D., Park C.Y., Kim Y.S. Word representation analysis of bio-marker and disease word // Advanced Science and Technology Letters. 2015. P. 793-797.
3. Liaw A. [et al.]. Classification and regression by random Forest // R news. 2002. Vol. 2, № 3. P. 18-22.
4. Bergstra J., Yamins D., Cox D.D. Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms // Proceedings of the 12th Python in science conference. 2013. P. 13-20.
5. Lerman P.M. Fitting segmented regression models by grid search // Journal of the Royal Statistical Society: Series C (Applied Statistics). 1980. Vol. 29, N 1. P. 77-84.
6. Люгер Д.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем, 4-е издание // Москва : ИД «Вильямс». 2003. С. 369-519.
7. Ketkar N. [et al.]. Deep Learning with Python // Apress. 2017. P. 159-194.
8. Hinton G.E., Srivastava N., Krizhevsky A., Sutskever I., Salakhutdinov R.R. Improving neural networks by preventing co-adaptation of feature detectors // arXiv preprint arXiv:1207.0580. 2012. P. 1-3.
9. Yingwei L., Sundararajan N., Saratchandran P. Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm // IEEE Transactions on neural networks. 1998. Vol. 9, N 2. P. 308-318.
10. Graves A., Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures // Neural Networks. 2005. Vol. 18, № 5-6. P. 602-610.
11. Николенко С., Кадурин А., Архангельская Е. Глубокое обучение. Погружение в мир нейронных сетей/ Санкт-Петербург : Питер. 2018. С. 6.
12. Аведьян Э.Д. Алгоритмы настройки многослойных нейронных сетей // Автоматика и телемеханика. 1995. № 4. С. 106-118.

Д. А. СТЕНЬКИН, В. И. ГОРБАЧЕНКО
Пензенский государственный университет
stynukin@mail.ru

РЕШЕНИЕ ОБРАТНЫХ КРАЕВЫХ ЗАДАЧ МАТЕМАТИЧЕСКОЙ ФИЗИКИ С ПОМОЩЬЮ СЕТЕЙ РАДИАЛЬНЫХ БАЗИСНЫХ ФУНКЦИЙ

Рассматривается задача нахождения неизвестного коэффициента уравнения Гельмгольца на основе значений заданного дополнительного условия. Предложен, реализован и исследован алгоритм решения задачи на сети радиальных базисных функций, обучаемой методом Левенберга–Марквардта.

Ключевые слова: обратная краевая задача, сеть радиальных базисных функций, обучение сети, метод Левенберга–Марквардта.

Введение

Среди краевых задач математической физики различают прямые и обратные краевые задачи [1]. При решении прямых краевых задач необходимо найти решение уравнения с частными производными с заданными граничными и начальными условиями. Лишь некоторые модельные задачи удаётся решить аналитически. В большинстве случаев уравнения решаются численно методами конечных разностей и конечных элементов. Новым направлением в решении краевых задач является бессеточный метод, реализуемый на специфическом виде нейронных сетей – сетях радиальных базисных функций (СРБФ) [2, 3]. Такой метод не требует построения сетки и позволяет получить приближенное дифференцируемое аналитическое решение в произвольных точках области.

В обратных задачах уравнение или его граничные или начальные условия заданы частично. Для того чтобы восстановить все компоненты используются некоторые известные дополнительные данные, полученные в результате каких-либо измерений в задачах реального мира [4]. Исходя из этого, можно классифицировать обратные задачи. В граничных обратных задачах необходимо восстановить граничные условия, в эволюционных обратных задачах необходимо восстановить начальные условия, и в коэффициентных обратных задачах необходимо восстановить некоторые коэффициенты уравнения.

Наиболее универсальным подходом к решению обратных задач является вариационный подход, согласно которому минимизируется функционал ошибки, представляющий собой сумму квадратов невязок между измерен-

ными в некоторых пробных точках характеристиками объекта и результатами моделирования в этих точках. Для минимизации функционала ошибки применяются градиентные методы. При использовании функциональной оптимизации градиент функционала ошибки вычисляется из решения сопряженной задачи, что представляет достаточно сложную задачу [1, 5]. При использовании параметрической оптимизации [4] применяется аппроксимация искомых параметров задачи в виде разложения по базисным функциям, что позволяет аналитически или численно вычислять градиент функционала. Перспективным направлением решения обратных краевых задач являются нейронные сети. Особенно перспективными являются СРБФ. Например, в [6] решена обратная задача для магнитного поля. Предложена РБФ-сеть, реализующая метод конечных элементов. На другой РБФ-сети корректируются параметры обратной задачи. В [7] решена коэффициентная обратная задача. Неизвестный коэффициент находится из решения системы нелинейных уравнений, полученных при использовании в качестве РБФ обратного квадрика. В работе [8] рассматривается граничная обратная задача. Для решения задачи используется РБФ-сеть. Параметры радиальных базисных функций задаются. Минимизация функционала ошибки осуществляется с помощью усеченного SVD-разложения.

Таким образом, в известных работах, посвященных решению обратных задач на СРБФ, сети, как правило, выполняют вспомогательные функции. Не используются СРБФ с настройкой всех параметров. Отсутствует единый подход к решению прямых и обратных задач. Как для отечественных, так и для зарубежных публикаций характерна недостаточная проработка вопросов обучения РБФ-сетей при решении обратных задач.

В [3, 9] предложен унифицированный нейросетевой подход к решению прямых и обратных задач. Подход использует параметрическую оптимизацию и СРБФ для аппроксимации как решения, так и параметров задачи, позволяет представить функционал ошибки в дифференцируемой приближенной аналитической форме и аналитически или численно вычислять градиент функционала ошибки.

При решении краевых задач на СРБФ чрезвычайно важно использование быстрых алгоритмов обучения сетей, так как решение задачи находится в процессе обучения нейронной сети. В [3] предложен быстрый алгоритм обучения СРБФ, основанный на методе доверительных областей, и рассмотрено решение коэффициентной обратной задачи для уравнения Гельмгольца. Но метод доверительных областей является достаточно сложным, так как на каждом шаге итерационного процесса обучения сети необходимо решать задачу условной оптимизации. В [10] для обучения СРБФ для решения краевых задач предложена адаптация метода Левенберга–Марквардта, соизмеримая по скорости обучения с методом доверительных областей, но более простая.

Целью данной работы является разработка и исследование алгоритма обучения СРБФ методом Левенберга-Марквардта при решении коэффициентной обратной задачи для уравнения Гельмгольца.

Анализ существующего подхода

СРБФ является двухслойной сетью [11]. Первый слой состоит из радиальных базисных функций (РБФ) φ, реализующих нелинейное преобразование входного вектора $\mathbf{x} = [x_1, x_2, ..., x_d]$ координат точки, в которой вычисляется приближение к решению, d – размерность пространства. Второй слой СРБФ представляет собой линейный взвешенный сумматор, описываемый выражением

$$u(\mathbf{x}) = \sum_{m=1}^{M} w_m \varphi_m(\mathbf{x}; \mathbf{p}_m), \quad (1)$$

где M – количество RBF, w_m – вес RBF φ_m, $\mathbf{p}_m$ – вектор параметров РБФ φ_m.

РБФ – это функции расстояния точки пространства от параметра функции, называемого центром функции: $\varphi(\|\mathbf{x}-\mathbf{c}\|, \mathbf{p})$, где $\mathbf{x}$ – точка пространства, $\mathbf{c}$ – центр радиальной базисной функции, $\|\mathbf{x}-\mathbf{c}\|$ – евклидова норма (расстояние) между точкой и центром, $\mathbf{p}$ – вектор параметров функции. Применяются различные RBF. В данной работе используется функция Гаусса (гауссиан), $\varphi(\|\mathbf{x}-\mathbf{c}\|, a) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{c}\|^2}{2a^2}\right)$, где $\mathbf{c}$ – координаты центра функции, a – параметр формы, часто называемый шириной.

Решение краевых задач на СРБФ рассмотрим на примере решения краевой задачи в операторном виде [3]:

$$Lu(\mathbf{x}) = f(\mathbf{x}),\ \mathbf{x} \in \Omega,\ Bu(\mathbf{x}) = p(\mathbf{x}),\ \mathbf{x} \in \partial\Omega, \quad (2)$$

где u – искомое решение; L – дифференциальный оператор, например, $L = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$; B – оператор граничных условий; Ω – область решения; $\partial\Omega$ – граница области; f и p – известные функции.

Обучение сети основано на минимизации функционала ошибки, представляющего собой сумму квадратов невязок в пробных точках для уравнения и граничных условий:

$$I(\mathbf{w}, \mathbf{p}) = \sum_{i=1}^{N} \left[Lu_{\text{RBF}}(\mathbf{x}_i; \mathbf{w}, \mathbf{p}) - f(\mathbf{x}_i)\right]^2 + \lambda \sum_{i=N+1}^{N+K} \left[Bu_{\text{RBF}}(\mathbf{x}_i; \mathbf{w}, \mathbf{p}) - p(\mathbf{x}_i)\right]^2, \quad (3)$$

где $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N \in \Omega$, $\mathbf{x}_{N+1}, \mathbf{x}_{N+2}, ..., \mathbf{x}_{N+K} \in \partial\Omega$; λ – подбираемый штрафной множитель, учитывающий вклад невязок в граничных пробных точках, u_{RBF} – приближение к решению (1), полученное СРБФ, N, K – количество пробных точек внутри и на границе области решения.

В операторном виде коэффициентную обратную задачу можно записать следующим образом [3]:

$$L(k(\mathbf{x},u))u(\mathbf{x}) = f(\mathbf{x}), \mathbf{x} \in \Omega, \tag{4}$$

$$Bu(\mathbf{x}) = p(\mathbf{x}), \mathbf{x} \in \partial\Omega, \tag{5}$$

где u – искомое решение; k – неизвестный коэффициент; L – дифференциальный оператор, зависящий от коэффициента; B – оператор граничных условий; Ω – область решения; $\partial\Omega$ – граница области; f и p – известные функции.

Дополнительное условие задано в виде:

$$Du(\mathbf{z}) = \psi(\mathbf{z}), \mathbf{z} \in Z, \tag{6}$$

где D – оператор дополнительных условий; $Z \subset \Omega \cup \partial\Omega$ – множество точек задания дополнительного условия; ψ – известная функция.

Используем параметрическую оптимизацию [1, 4]. Для параметрического представления искомого коэффициента используем СРБФ

$$k_{RBF}(\mathbf{x},u) = \sum_{m=1}^{M_k} w_m^k \varphi_m^k(\mathbf{x},u;\mathbf{p}_m^k), \tag{7}$$

где M_k – количество РБ-функций, w_m^k, $\mathbf{p}_m^k$ – веса и параметры РБФ, φ_m^k – РБФ, зависящая как от x, так и от u.

Неизвестные параметры $\mathbf{p}_m^k$ и веса РБФ w_m^k выбираются из условия минимизации невязки между левой и правой частями (6)

$$J^k(\mathbf{w}^k, \mathbf{p}^k) = \sum_{s=1}^{S} \left[Du(\mathbf{z}_s) - \psi(\mathbf{z}_s) \right]^2, \tag{8}$$

где S – количество контрольных точек из Z, $\mathbf{w}^k = \left[w_1^k, w_2^k, .. w_{M_k}^k \right]$, $\mathbf{p}^k = \left[\mathbf{p}_1^k, \mathbf{p}_2^k, .. \mathbf{p}_{M_k}^k \right]$.

Обратные задачи являются некорректными, т.е. существует множество решений, удовлетворяющих с некоторой точностью (8). Для выбора однозначного решения использована итерационная регуляризация Морозова [12] – итерационный процесс минимизации прекращается, когда перестает уменьшаться (8).

Неизвестное решение u прямой задачи (4)–(5), в которой $k = k_{RBF}$ аппроксимируется с помощью СРБФ $u_{RBF}(\mathbf{x}) = \sum_{m=1}^{M_u} w_m^u \varphi(\mathbf{x}; \mathbf{p}_m^u)$, где M_u – количество РБФ φ_m^u, w_m^u, $\mathbf{p}_m^u$ – веса и параметры РБФ.

Неизвестные параметры $\mathbf{p}_m^u$ и веса РБФ w_m^u можно найти, минимизировав функционал

$$J^u(\mathbf{w}^u, \mathbf{p}^u) = \sum_{i=1}^{N} \left[L(k_{RBF}(\mathbf{x}_i, u_{RBF}(\mathbf{x}_i))) u_{RBF}(\mathbf{x}_i) - f(\mathbf{x}_i) \right]^2 + \\ + \lambda_B \sum_{i=N+1}^{N+K} \left[B u_{RBF}(\mathbf{x}_i) - p(\mathbf{x}_i) \right]^2, \tag{9}$$

где $\mathbf{w}^u = \left[w_1^u, w_2^u, .. w_{M_u}^u \right]$, $\mathbf{p}^u = \left[\mathbf{p}_1^u, \mathbf{p}_2^u, .. \mathbf{p}_{M_u}^u \right]$, λ_B – штрафной множитель за нарушение граничных условий (5).

Объединение функционалов (8) и (9) приходит к задаче минимизации функционала^

$$J(\mathbf{w}^u, \mathbf{p}^u, \mathbf{w}^k, \mathbf{p}^k) = J^u(\mathbf{w}^u, \mathbf{p}^u) + \lambda_D J^k(\mathbf{w}^k, \mathbf{p}^k), \tag{10}$$

где λ_D – штраф за нарушение дополнительных условий (8).

Решение коэффициентной обратной задачи на СРБФ, обучаемой методом Левенберга–Марквардта

Коррекция вектора θ параметров СРБФ, составленного из весов и параметров РБФ, на итерации k при решении задачи (2) производится по формуле $\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} + \Delta\boldsymbol{\theta}^{(k+1)}$, в которой вектор поправки параметров $\Delta\boldsymbol{\theta}^{(k)}$ находится из решения системы линейных алгебраических уравнений:

$$\left(\mathbf{J}_{k-1}^{\mathrm{T}} \mathbf{J}_{k-1} + \mu_k \mathbf{E} \right) \Delta\boldsymbol{\theta}^{(k)} = -\mathbf{g}_{k-1}, \tag{11}$$

где $\mathbf{J}_{k-1}$ и $\mathbf{J}_k$ – матрицы Якоби, вычисленные в $k-1$ и k итерациях, $\mathbf{E}$ – единичная матрица, μ_k – параметр регуляризации, изменяющийся на каждом шаге обучения, $\mathbf{g} = \mathbf{J}^{\mathrm{T}}\mathbf{r}$ – вектор градиента функционала (3) по вектору параметров $\boldsymbol{\theta}$, $\mathbf{r}$ – вектор невязок во внутренних и граничных пробных точках.

Матрица Якоби представляет собой матрицу производных вектора невязок $\mathbf{r}$ по элементам вектора параметров θ и для СРБФ несложно вычисляется аналитически.

Рассмотрим решение коэффициентной обратной задачи на примере уравнения Гельмгольца. Прямая задача для этого уравнения имеет вид

$$-\frac{\partial^2 u(x,y)}{\partial x^2}-\frac{\partial^2 u(x,y)}{\partial y^2}+c(y)u(x,y)=0,\qquad (x,y)\in\Omega, \tag{12}$$

$$u(x,y)=p(x,y),\ (x,y)\in\partial\Omega, \tag{13}$$

где Ω – расчетная область; $\partial\Omega$ – граница расчетной области; $c=10y$, $p=1+x$. Расчётной областью является квадрат 1×1.

Дополнительное условие имеет вид

$$\frac{\partial u(x,y)}{\partial \mathbf{n}}=\psi(x,y),\quad (x,y)\in\partial\Omega, \tag{14}$$

где $\mathbf{n}$ – внешняя по отношению к границе Ω нормаль.

В реальной задаче значения $\psi(x,y)$ получаются с некоторой погрешностью в результате измерений на границе Ω. В модельной задаче эти значения вычислялись по результатам решения прямой задачи (12) с известным коэффициентом $c=10y$. Прямая задача решалась на СРБФ. Использование для обучения сети метода градиентного спуска показало очень низкую скорость сходимости. Исследовано применение для обучения сети метода Левенберга–Марквардта. Однако при использовании метода Левенберга–Марквардта матрица системы (11) быстро приближается к диагональной матрице с очень большими диагональными элементами. В результате вектор поправки практически равен нулю и сеть плохо обучается.

Для преодоления этого недостатка уравнение (12) рассмотрено как уравнение Пуассона с нелинейной правой частью. Уравнение решалось итерационным методом $\Delta u^{(n+1)}=c\left(u^{(n)}\right)\cdot u^{(n)}$, где Δu – Лапласиан, n – номер итерации по нелинейности. На каждой итерации по нелинейности задача решалась на СРБФ методом Левенберга–Марквардта.

График численного решения с точным коэффициентом показан на рис. 1.

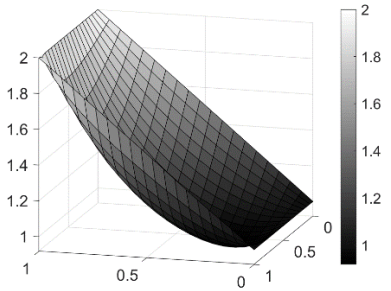


Рис. 1. График численного решения с точным коэффициентом

Точки задания дополнительных условий располагались случайно на границе области (рис. 2).

Дифференцируемость решения прямой задачи, полученного на RBF-сети, позволяет вычислить нормальные производные

$$\frac{\partial u}{\partial n} = -\sin\alpha\left(\sum_{k=1}^{n_{RBF}} \frac{w_k}{a_k^2} e^{-\frac{\|\mathbf{x}-\mathbf{c}_k\|^2}{a_k^2}} \cdot (x_1 - c_{k1})\right) - \cos\alpha\left(\sum_{k=1}^{n_{RBF}} \frac{w_k}{a_k^2} e^{-\frac{\|\mathbf{x}-\mathbf{c}_k\|^2}{a_k^2}} \cdot (x_2 - c_{k2})\right), \quad (15)$$

где α – угол между направлением нормали и осью абсцисс, $\mathbf{x}$ – вектор координат точки задания дополнительных условий, x_1, x_2, c_{k1} и c_{k2} – координаты точки задания дополнительных условий и РБФ.

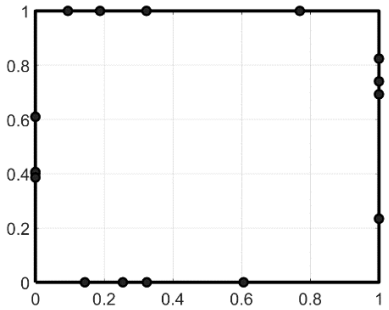


Рис. 2. Положение точек дополнительного условия

Предложенный подход не позволяет напрямую минимизировать функционал (8). Предлагается следующий алгоритм:

Инициализация сетей и коэффициента c

while $J^{(n)} > \varepsilon$ **do**

Решение прямой задачи $\Delta u^{(n)} = c^{(n-1)} u^{(n-1)}$

Расчет дополнительных условий

Одна итерация пересчета коэффициента

end while

Алгоритм реализует минимизацию функционала (8) до точности, определяемой точностью измерения дополнительных условий в реальной задаче. Тем самым реализуется итерационная регуляризация. Решение прямой задачи – уравнения Пуассона с нелинейной правой частью – производится на СРБФ методом Левенберга–Марквардта. Одна итерация пересчета коэффициента производится на другой сети методом градиентного спуска. Градиент функционала (8) по параметрам этой сети вычисляется путем численного дифференцирования, так как аналитическое выражение зависимости (8) от параметров этой сети отсутствует.

Анализ результатов решения модельной задачи

Число используемых нейронов для решения прямой задачи равно 64. Начальное значение вектора весов равно 0. Начальные значения компонентов вектора весов равны 0.2. Центры располагались регулярно на квадратной сетке размерами 8×8. Обучение СРБФ проводилось методом Левенберга–Марквардта. Значение функционала ошибки, равное 0.0001, достигается в среднем за 58 итераций.

Для пересчета приближения к искомому коэффициенту (7) использовалась СРБФ с числом нейронов $nRBFc = 10$. В качестве РБФ использовались функции Гаусса. В процессе обучения сети минимизировался функционал ошибки (8), в котором $Du(\mathbf{z}_s)$ и $\psi^{\delta}(\mathbf{z}_s)$ вычисляются аналитически по (15). Обучение сети проводилось методом градиентного спуска. Градиент функционала (8) вычислялся с использованием численного дифференцирования решения прямой задачи в точках задания дополнительных условий. Первоначально центры располагались на отрезке равномерно с шагом $h = 1/(nRBFc-1)$. Начальная ширина равна h. Скорости обучения весов, центров и ширины подобраны экспериментально и, соответственно, равны $\eta_w = 0.001, \eta_c = 0.001, \eta_a = 0.0001$.

Функционал (8) практически перестал изменяться после 775 итераций. При этом достигнута средняя квадратическая погрешность восстановленного коэффициента относительно точного значения

$$\varepsilon^c = \sqrt{\sum_{i=1}^{S}(c_{RBF,i}-c_i)^2} \Big/ \sqrt{\sum_{i=1}^{S} c_i^2} = 0.005,$$

где S – количество точек задания дополнительных условий, $c_{RBF,i}$ – значение восстановленного коэффициента, c_i – точное значение коэффициента, i – номер точки задания дополнительных условий.

Погрешность также оценивалась по формуле средней квадратической погрешности решения прямой задачи с восстановленным и точным коэффициентами

$$\varepsilon^u = \sqrt{\sum_{i=1}^{N+K}(u_{RBFc_{RBF},i}-u_{RBFc,i})^2} \Big/ \sqrt{\sum_{i=1}^{N+K} u_{RBFc,i}^2} = 0.0007,$$

где N, K – количество пробных точек внутри и на границе области решения, $u_{RBFc_{RBF},i}$ – решение прямой задачи с восстановленным коэффициентом, $u_{RBFc,i}$ – решение прямой задачи с точным коэффициентом, i – номер точки внутри и на границе.

График сравнения точного коэффициента и коэффициента, полученного на СРБФ показан на рис. 3. Графики близки.

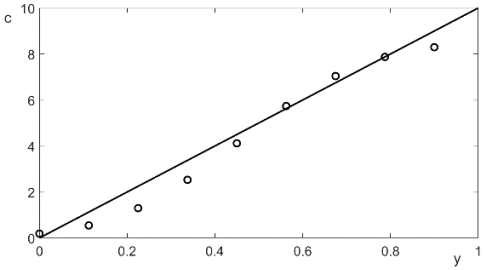


Рис. 3. Сравнение коэффициентов, полученных на сети с точным значением

График численного решения с коэффициентом, полученным после решения обратной задачи, практически не отличается от графика решения с точным коэффициентом.

Выводы

Предложен, реализован и исследован алгоритм на основе сетей радиальных базисных функций для нахождения неизвестного коэффициента уравнения Гельмгольца на основе значений заданного дополнительного условия. В качестве алгоритма решения прямой задачи был выбран алгоритм решения на сети, обучаемой методом Левенберга–Марквардта. Относительная среднеквадратическая погрешность восстановления искомого коэффициента равна 0.005. Относительная среднеквадратическая погрешность решения с восстановленным коэффициентом равна 0.0007. Таким образом, эксперименты с программами показали, что была достигнута требуемая точность аппроксимации.

Список литературы

1. Самарский А.А., Вабищевич П.Н. Численные методы решения обратных задач математической физики. Москва : Изд-во ЛКИ. 2009. 480 с.
2. Yadav N., Yadav A., Kumar M. An Introduction to Neural Network Methods for Differential Equations. Springer, 2015. 115 p.
3. Горбаченко В.И., Жуков М.В. Решение краевых задач математической физики с помощью сетей радиальных базисных функций // Журнал вычислительной математики и математической физики. 2017. Т. 57, № 1. С. 133–143.
4. Aster R.A., Borchers B., Thurber C.H. Parameter estimation and inverse problems. Elsevier. 2013. 356 p.
5. Kern M. Numerical Methods for Inverse Problems. Wiley. 2016. 232 p.
6. Finite-Element Neural Network-Based Solving 3-D Differential Equations in MFL // Xu C., Wang C., Ji F., Yuan X. // IEEE Transactions on Magnetics. 2012, Vol. 48, N 12. P. 4747–4756.
7. Parzlivand F., Shahrezaee A. The use of inverse quadratic radial basis functions for the solution of an inverse heat problem // Bulletin of the Iranian Mathematical Society. 2016, Vol. 42, N 5. P. 1127–1142.
8. Li Z., Mao X.-Z. Least-square-based radial basis collocation method for solving inverse problems of Laplace equation from noisy data // International Journal for Numerical

Methods in Engineering. 2010. Vol. 84, Issue 1. P. 1–26.
9. Neural Network Technique in Some Inverse Problems of Mathematical Physics // Gorbachenko V.I. , Lazovskaya T.V., Tarkhov D.A., Vasiljev A.N., Zhukov M.V. // Advances in Neural Networks - ISNN 2016: 13th International Symposium on Neural Networks, ISNN 2016, St. Petersburg, Russia, July 6-8, 2016, Proceedings (Lecture Notes in Computer Science). Springer, 2016. P. 310–316.
10. Горбаченко В.И. Обучение сетей радиальных базисных функций методом Левенберга–Марквардта при решении краевых задач // XVII Всероссийская научная конференция "Нейрокомпьютеры и их применение". Тезисы докладов. Москва : МГППУ, 2019. С. 324–326.
11. Aggarwal C.C. Neural Networks and Deep Learning. Springer, 2018. 497 p.
12. Морозов В.А. Регулярные методы решения некорректно поставленных задач. Москва : Наука, 1987. 240 с.

Научное издание

XXI МЕЖДУНАРОДНАЯ НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ "НЕЙРОИНФОРМАТИКА-2019"

Сборник научных трудов
В 2-х частях
Часть 1

Редакторы: *В.А. Дружинина, О. П. Котова*
Корректоры: *О. П. Котова, В.А. Дружинина*
Компьютерная верстка *Д. А. Юдин*

Подписано в печать 30.09.2019. Формат 60 × 84 $^{1}/_{16}$. Усл. печ. л. 12,4.
Уч.-изд. л. 11,8. Тираж 200 экз. Заказ № 162.
Федеральное государственное автономное образовательное учреждение
высшего образования «Московский физико-технический институт
национальный исследовательский университет)»
141700, Московская обл., г. Долгопрудный, Институтский пер., 9
Тел. (495) 408-58-22, e-mail: rio@mipt.ru

Отпечатано в полном соответствии с предоставленным оригиналом-макетом
ООО «Печатный салон ШАНС» 127412, г. Москва, ул. Ижорская, д. 13, стр. 2