

УДК 519.68, 681.3.06

## **Вопросы реализации адаптивного генерирования электронных учебных курсов в интеллектуальных обучающих системах**

*Волянская Т.А. (Институт систем информатики СО РАН)*

Статья посвящена вопросам адаптации в интеллектуальных обучающих системах, а именно методам и средствам реализации адаптивного генерирования электронных учебных курсов. Рассматриваются интеллектуальные системы обучения с адаптивным генерированием электронных учебных курсов, предоставляющих индивидуализированный учебный материал для каждого учащегося и динамически выбирающих оптимальный метод обучения на каждом шаге процесса обучения, а также используемые в них методы и средства адаптации.

***Ключевые слова:** электронное обучение, интеллектуальные системы обучения, адаптивные обучающие системы, обучающие гипермедиа-системы, адаптивное генерирование электронных учебных курсов, адаптивная гипермедиа, адаптивное представление, адаптивная навигация, графы планов обучения, графы путей концептов, графы учебной деятельности, концептуальные карты, онтологии, байесовские сети.*

### **1. Введение**

В последнее время электронное обучение (E-learning), под которым понимается система обучения при помощи информационных и электронных технологий, также называемое онлайн-обучением (Online learning), или цифровым образованием (Digital learning), становится важной частью образовательной системы.

Электронное обучение в наше время активно применяется в школах, вузах, при дополнительном образовании. Использование цифровых технологий для улучшения процесса преподавания и обучения предоставляет новые возможности для педагогов и учащихся и охватывает широкий спектр инструментов и методов, включая онлайн-курсы, виртуальные классы, интерактивные учебные материалы, электронные учебные курсы и многое другое. Всё более популярным становится такой инструмент обучения, как массовые открытые онлайн-курсы.

Мир переживает цифровую трансформацию образования, помимо широкого распространения цифровых технологий, интернета и смартфонов, виртуальной реальности и искусственного интеллекта, свою роль здесь сыграла пандемия коронавируса и связанный с ней переход на дистанционное обучение. Смешанное, гибридное или полностью дистанционное обучение стало теперь для многих привычным [3].

Преимуществом традиционной формы образования является возможность преподавателя адаптировать процесс обучения к учащимся, принимая во внимания их индивидуальные особенности. Один из основополагающих принципов дидактики – индивидуализация обучения, от реализации которого во многом зависит эффективность обучения. Под индивидуализацией обучения понимают организацию учебного процесса с учетом психолого-педагогических характеристики учащихся, таких как особенности мышления, памяти и внимания, уровень развития интеллектуальных способностей и общеучебных умений и навыков, уровень усвоения программного материала [1].

Одна из ключевых тенденций цифрового образования – персонализация обучения. Персонализация обучения позволяет создавать адаптивные учебные программы, которые могут быть настроены под индивидуальные потребности и интересы каждого учащегося, предоставляя персонализированные рекомендации и учебные материалы, соответствующие уровню знаний и интересам учащихся. Применение технологий искусственного интеллекта и машинного обучения для создания персонализированных адаптивных учебных курсов и рекомендаций позволяет значительно повысить эффективность обучения и мотивацию учащихся. Цифровая образовательная среда ставит перед дидактикой новые задачи внедрения современных технологий, методов и средств электронного обучения, чтобы сделать цифровое образование более персонализированным [5,64].

Интеллектуальные обучающие системы, активно развивающиеся в последнее время, являются одним из эффективных инструментов электронного обучения, в связи с этим, направления исследований методов и средств персонализации и адаптации в системах обучения как никогда актуальны. Адаптивные гипермедиа-системы значительно увеличивают возможности интеллектуальных обучающих систем, их целью является персонализация гипермедиа-системы, ее настройка на индивидуальные особенности пользователей [7,8,11,13,21,23,32,37,38,39,41,64,72-75].

В крупных информационных гипермедиа-системах, характеризующимся достаточно обширным гиперпространством и предусматривающих работу с множеством пользователей с различными целями, уровнем знаний и опытом, становится целесообразным применение методов адаптации. В связи с этим, технологии адаптивной гипермедиа используются не

только в системах электронного обучения, а также, например, в информационно-поисковых системах, информационно-справочных системах, электронных библиотеках и архивах, а также открытых адаптивных виртуальных музеях [9,10,12,14,18-20,22,24,69-71].

В настоящее время в связи с высоким темпом развития образовательной системы, в результате активного роста электронных информационно-образовательных ресурсов и сетевых дистанционных образовательных технологий возрастает необходимость разработки и реализации адаптивных электронных учебных курсов, предоставляющих индивидуализированный учебный материал для каждого учащегося и динамически выбирающих оптимальный метод обучения на каждом шаге процесса обучения. В связи с тем, что одной из главных причин развития и широкого распространения электронного обучения является именно рост знаний учащихся, более важными становятся вопросы адаптации электронных учебных курсов, касающиеся адаптации и упорядочения учебных материалов, чем вопросы адаптивного представления, типичные для адаптивных обучающих гипермедиа-систем [7,8,11,64].

Более тридцати лет назад в системах электронного обучения уже начала применяться простейшая адаптация, но до сих пор эталонная модель генерирования электронных учебных курсов не была определена. В настоящее время в научной литературе встречается множество различных терминов, употребляемых в значении генерирования электронного учебного курса (courseware generation): упорядочение учебного курса (courseware sequencing), упорядочение плана обучения (curriculum sequencing), генерирование маршрута (trail generation), планирование учебного курса (course planning), учебное моделирование (instructional modelling), упорядочение внутри учебных модулей (sequencing within learning modules) и учебное планирование (instructional planning) [64,97].

Также встречаются такие термины и понятия, как динамическое, адаптивное, персонализированное и интеллектуальное генерирование электронного учебного курса (dynamic, adaptive, personalized and intelligent courseware generation), используемые в значении генерирования электронного учебного курса, учитывающего и отвечающего потребностям учащегося [47].

В современных интеллектуальных системах обучения применяются следующие методы адаптации: упорядочение контента (content sequencing), интеллектуальный анализ решений учащихся (intelligent analysis of students' solutions), интерактивная поддержка решения задач (interactive problem solving support), поддержка решения задач на основе примеров (example-based support for problem solving) и поддержка совместной работы (collaboration support). Методы адаптации, применяемые в современных адаптивных обучающих гипермедиа-

системах, это адаптивное представление (adaptive presentation) и адаптивная навигационная поддержка (adaptive navigation support) [7,11,13,37,38,64].

Данная статья посвящена исследованию методов и средств реализации адаптивности в интеллектуальных системах обучения, а именно вопросам адаптивного генерирования электронных учебных курсов [7,8,11].

В первом разделе статьи рассматриваются вопросы, относящиеся к генерированию адаптивных электронных учебных курсов, предоставляющих индивидуализированный учебный материал для каждого учащегося и динамически выбирающих оптимальный метод обучения на каждом шаге процесса обучения. Приводятся следующие методы, применяемые для реализации адаптации электронных учебных курсов: графы путей концептов, графы планов обучения, графы учебной деятельности, концептуальные карты, онтологии, байесовские сети.

Второй раздел статьи посвящен обзору интеллектуальных систем обучения, автоматически или динамически генерирующих адаптивные электронные учебные курсы, с акцентом на адаптацию контента, а не интерфейса. Рассматриваются вопросы формализации предметной области и модели обучаемого в этих системах. Описываются процесс и этапы генерирования адаптивных электронных учебных курсов в данных системах: генерирование, выбор и упорядочение, представление элементов учебного курса. Рассматриваются функции адаптации и тестирования знаний учащихся [64].

## **2. Методы генерирования адаптивных электронных учебных курсов**

Одним из наиболее популярных и распространенных методов адаптации в интеллектуальных системах обучения является генерирование электронных учебных курсов. Электронный учебный курс состоит модулей двух типов: обучающих модулей и модулей для проверки знаний. Под генерированием электронного учебного курса понимается процесс выбора модулей курса и их упорядочения подходящим образом для определенной группы или отдельных учащихся. Смысл адаптивного генерирования электронного учебного курса заключается в том, чтобы генерировать индивидуализированный учебный материал для конкретного учащегося и динамически выбирать оптимальный метод обучения на каждом шаге процесса обучения. Под оптимальным методом обучения понимается тот, который приведет учащегося к реализации его цели обучения. Самая распространенная цель обучения – это приобретение необходимых знаний в кратчайший срок. Это одно из наиболее

перспективных направлений научных исследований в области электронного обучения [8,41,67,82].

Процесс генерирования электронных учебных курсов в интеллектуальных системах обучения включает в себя два основных этапа: планирование учебного материала (контента) (content planning) и планирование представления учебного материала (контента) (content delivery planning) [51].

Планирование учебного материала включает в себя следующие шаги: генерирование, упорядочение и выбор элементов контента, основываясь на текущем знании учащегося, и наблюдении за выполнением плана контента для определения того, когда следует перепланировать (адаптировать уже существующий план) или генерировать новый план.

Планирование представления учебного материала, также называемое стратегией обучения, относится к выбору действий и взаимодействий, которые помогут учащемуся достичь целей обучения. Самые распространенные стратегии обучения, это обучение на примере (learning by example), обучение с помощью чтения текстов (learning by reading texts), обучение на собственном опыте (learning by doing) и моделирование (simulation) [106].

В современных интеллектуальных системах обучения используются различные подходы к генерированию электронных учебных курсов. Большинство систем генерируют электронные учебные курсы, используя только один метод обучения. Некоторые системы способны изменять только порядок вопросов или задач, другие – располагать по порядку уроки, но более прогрессивные системы могут генерировать учебный курс, используя разнообразные методы обучения.

Существует три подхода к генерированию электронных учебных курсов: статическое, адаптивное и динамическое генерирование. Традиционные статические обучающие системы адаптируют процесс обучения, используя подходящие педагогические стратегии для представления знаний предметной области. Модель учебного курса в большинстве таких систем – статический массив или дерево, при этом все элементы учебного курса создаются преподавателем заранее, а курс генерируется сразу же, что снижает его адаптивность [88].

Идея адаптивного генерирования учебного курса заключается в создании специализированного содержания до того, как учащиеся начнут его использовать. В случае динамического генерирования учебного курса система отслеживает прогресс обучаемого, и в случае несоответствия его успеваемости ожидаемой, учебный материал динамически регенерируется. При этом подходе принимаются во внимание текущие знания, цели и период обучения учащихся, и соответственно изменяется уровень сложности и происходит адаптация к прогрессу (успеваемости) учащихся [41].

Во всех системах обучения, в которых применяются методы генерирования электронных учебных курсов, присутствуют т.н. элементы учебного курса: элементы знаний предметной области, концепты, фрагменты знаний, темы, учебные модули, объекты знаний (domain knowledge elements, concepts, fragments of knowledge, themes, learning objects, knowledge objects) [110].

В настоящее время в интеллектуальных системах обучения применяются следующие методы адаптации учебных курсов: графы путей концептов, графы планов обучения, графы учебной деятельности, концептуальные карты, онтологии, Байесовские сети [34].

## **2.1. Графы путей концептов, планов обучения и учебной деятельности**

**Граф путей концептов (Concept Path Graph).** Это ориентированный ациклический граф, который представляет собой множество правил упорядочения, определяющих последовательность концептов, снабженных набором характеристик, которые должен приобрести учащийся. Под концептом понимается элемент знаний предметной области, для каждого концепта в графе создается вершина, которая его представляет. Для определения структуры пути концептов на основе модели предметной области и модели пользователя используются учебные цели. Таким образом, граф путей концептов представляет концептуальную структуру онтологии предметной области, соответствующей рассматриваемой цели обучения. Концепты, содержащиеся в графе путей концептов, выбираются согласно взаимосвязям между иерархией учебных целей и онтологией предметной области. Структура графа полностью копирует структуру онтологии предметной области [44,68,92].

**Граф планов обучения (Learning Path Graph).** Это ориентированный ациклический граф, который определяет все возможные планы обучения (последовательность учебных материалов), соответствующие данной цели обучения. Граф планов обучения описывает структуру предметной области и связанные с ней учебные цели, ставя в соответствие учебную цель и все возможные для ее достижения планы обучения. Основываясь на характеристиках учащегося в модели обучаемого, из этого графа, содержащего все возможные планы обучения, выбирается индивидуальный план обучения. Модель обучаемого содержит информацию об уровне знаний учащегося, а также характеристиках, таких как стиль обучения и предпочтения. Проектирование модели предметной области включает два этапа: проектирование иерархии задач (учебных целей) и составление набора понятий (концептов) [67,68].

Чтобы построить граф планов обучения, для каждого концепта из графа путей концептов из медиа пространства выбираются связанные с ним учебные ресурсы на основе связи между онтологией предметной области и моделью описания ресурсов. Медиа пространство содержит описания характеристик учебных материалов. Каждый узел в графе путей концептов далее замещается связанным с ним подмножеством учебных материалов, извлеченным из медиа пространства. Структура множества учебных материалов копирует непосредственно структуру медиа пространства. Конечный граф является графом планов обучения. При условии, что медиа пространство не содержит циклических ссылок между учебными материалами, граф планов обучения становится простым бесконтурным орграфом. Хотя это предположение не влияет прямым образом ни на проектирование системы, ни на методологию упорядочения, необходимо избегать бесконечных планов обучения.

На следующем этапе из графа, содержащего все доступные планы обучения, выбирается персонализированный план обучения, основываясь на характеристиках модели обучаемого. В итоге получается дополнительный уровень, называемый уровнем адаптации обучаемого, который используется для выбора персонализированного плана обучения [44,46].

**Граф учебной деятельности (Learning Activity Graph).** Это ориентированный граф, который используется для организации учебных ресурсов, связанных с учебной задачей. Общая учебная задача может быть представлена в виде иерархического графа учебных действий, где вершины графа могут быть либо простым учебным действием, либо подграфом учебных действий, включающим несколько связанных учебных ресурсов для учебной подзадачи. В данном случае простым учебным действием считается учебный ресурс, содержащий учебный материал, который должен быть представлен учащемуся обучающей системой.

Граф учебной деятельности определяется следующим образом:  $G = (V, E, v_i, v_0)$ , где  $V$  – множество вершин, каждая из которых является учебным ресурсом, который может посетить учащийся, с пред- и пост- условиями для определения следующей вершины;  $E$  – множество дуг, каждая из которых соединяет некоторые две вершины из  $V$ ;  $v_i$  – начальная вершина, а  $v_0$  – конечная вершина графа,  $v_i, v_0 \in V$ . Структура вершины определяется следующим образом:

preCondition	nodeType	learningResource	postCondition	startNode	endNode
--------------	----------	------------------	---------------	-----------	---------

В этой структуре `nodeType` обозначает тип вершины, т.е. простое действие или группу действий, `learningResource` – учебный ресурс, соответствующий действию, `preCondition` и `postCondition` обозначают соответственно пред- и пост- условия, `startNode` и `endNode` обозначают соответственно начальную и конечную вершины в (под)графе учебных действий. Таким образом, дуга  $e \in E$  является тройкой  $(v_s, v_c, C)$ , где  $v_s$  и  $v_c$  – начальная и

конечная вершины дуги,  $v_s, v_c \in V$ ; а *preCondition* или *postCondition* *C* – логические выражения. Если условие *C* оценивается как истинное, будет выбрано соответствующее упорядочивание. При этом, если таких дуг существует несколько, одна из них выбирается случайным образом [111].

## 2.2. Концептуальные карты

Концептуальная карта (*Concept Map*) – также концепт-карта, понятийная карта, диаграмма концептов, представляет собой графический инструмент для структурирования и представления знаний предметной области.

Концептуальная карта состоит из концептов (понятий), обычно изображаемых в фигурах определенного типа (кругов или прямоугольников), и отношений между понятиями, изображаемых соединительными линиями, связывающими два понятия. Слова на этих линиях, называемые связующими словами или связующими фразами, определяют связь между этими двумя понятиями. Концептуальные карты помогают показать, как человек видит взаимосвязи между понятиями, идеями, объектами, явлениями и др. [84-86].

Математически концептуальная карта представляет собой ориентированный помеченный граф, вершинам которого соответствуют концепты (понятия), дугам – отношения (связи) между понятиями, при этом метками на дугах служат связующие фразы.

**Определение.** Концепт, согласно определению, данному Джозефом Новаком (*Joseph Novak*), основоположником концептуальных карт, есть «ментальный образ предмета, объекта или абстрактной идеи». Таким образом, концепт определяется как воспринимаемая закономерность или модель объектов или явлений, или данные об объектах или явлениях, обозначенный меткой [85].

Иначе говоря, концепты – это короткие формулировки, чаще состоящие из одного слова, связанные с понятиями предметной области, описывающими исследуемую ситуацию, или закономерностями в объектах, событиях и ситуациях.

На соединительных линиях, связывающих два концепта (понятия), указывают связующие слова или фразы, определяющие и разъясняющие тип связи между понятиями (иногда используются символы: +, -, % и др.). Эти слова-отношения чаще всего являются глаголами.

Каждые два концепта со связующим словом-отношением образуют короткое предложение, называемое пропозицией (*proposition*). Пропозиция – это осмысленное высказывание, являющееся семантической единицей или смысловой единицей. В некоторых случаях пропозиция может содержать три и более концепта, но, как правило, такой ситуации

стараятся избегать, поскольку утверждения могут стать размытыми и неопределенными [81].

**Основные характеристики.** Основная идея концептуальных карт состоит в том, что взаимосвязанность является важнейшим свойством знаний, и что «понимание» знаний может быть представлено посредством широкого набора взаимосвязей между важными концептами предметной области.

*Семантическая структура.* Два (или более) концепта, объединенные посредством связующей фразы в осмысленное высказывание, составляют «пропозицию» – смысловую (или семантическую) единицу. Согласно когнитивной теории Дэвида Озубела (David Ausubel), концепты (понятия) и пропозиции (высказывания), будучи наименьшими единицами знаний, служат основой для создания новых знаний в предметной области. Можно провести аналогию с тем, что концепты подобны атомам, а пропозиции – молекулам. Концептуальные карты с помощью связующих фраз наглядно отображают наиболее важные взаимосвязи между набором концептов, формируя пропозиции. По сути, концептуальная карта визуально передает набор утверждений на заданную тему. При составлении концептуальной карты нужно следить за тем, чтобы каждые два понятия вместе со связывающими их фразами образовывали смысловую единицу, утверждение, короткое предложение [31].

*Иерархическая структура.* В любой области знаний существует иерархия понятий, где наиболее общие понятия находятся на «вершине» иерархии, а более конкретные расположены иерархически ниже. Одна из ключевых характеристик концептуальных карт – иерархичность ее структуры: наиболее общие, содержательные и всеобъемлющие концепты находятся на верхних уровнях иерархии, а менее общие, более конкретные и специфичные концепты расположены на нижних уровнях. Соответственно, иерархические концептуальные карты читаются сверху вниз. Тем не менее, структура концептуальной карты не ограничивается этой структурой, она может принимать и свободную форму – начиная от центра и распространяясь наружу.

*Ключевой вопрос.* Концептуальные карты создаются на основе ключевого вопроса – формулировки, которая задает и четко определяет основную тему или проблему, которую карта предназначена решить. Ключевой вопрос помещается в самом верху концептуальной карты и служит отправной точкой для дальнейших ответвлений. Хороший ключевой вопрос может привести к созданию гораздо более подробной концептуальной карты.

*Перекрестные ссылки.* Другой важной характеристикой концептуальных карт является включение перекрестных ссылок (связей). Поскольку концептуальные карты содержат

концепты из различных областей знаний, отношения между этими областями отображаются с помощью перекрестных ссылок. Это связи или отношения между концептами из разных сегментов или областей концептуальной карты. Такие перекрестные ссылки помогают понять, как концепт из одной области знаний, представленной на карте, связан с концептом из другой области, показанной на карте, и, таким образом, иллюстрирует связь этих частей друг с другом.

*Конкретные примеры.* Последняя особенность, которая может быть присуща концептуальным картам, это конкретные примеры объектов или явлений, помогающих разъяснить значение данного концепта. Обычно они не заключены в овалы или прямоугольники, поскольку являются конкретными объектами или явлениями и не являются концептами.

**История создания.** Концептуальные карты впервые были предложены в 1960-1970-х годах американским педагогом и ученым-исследователем из Корнельского университета (США) Джозефом Новаком (Joseph D. Novak), который занимался изучением детского мышления и формированием первых научных понятий. Для изучения того, как дети понимают концепции фундаментальных наук, были проведены и обработаны интервью со многими детьми. Необходимость найти лучшее решение для представления концептуального понимания детей привела к разработке концептуальных карт. С тех пор они находят применение не только в сфере образования и научных исследований, но также и в бизнесе [84-86].

Исследования Джозефа Новака основывались на идеях американского когнитивного психолога и педагога Дэвида Озубела (David Ausubel), занимавшегося вопросами обучения студентов в колледжах и формированием понятийного мышления. В начале 1960-х годов им была разработана когнитивная теория ассимиляции, впоследствии широко применяемая в области осмысленного вербального обучения. В основе этой теории лежит тот факт, что новые знания можно эффективно усваивать, связывая их с тем, что уже известно, при этом концептуальные карты рассматриваются как методологический инструмент этой теории [30].

Согласно теории Дэвида Озубела, обучение людей, независимо от их возраста, зависит от их предыдущей когнитивной структуры, которая связана с новой информацией. Под когнитивной структурой понимаются ментальные репрезентации (набор идей и концепций), которые человек строит о фрагменте реальности. Иначе говоря, это совокупность всех полученных знаний, а также взаимосвязей между фактами, понятиями и принципами, из которых состоят эти знания. По мнению Дэвида Озубела, обучение заключается в том, чтобы

привнести что-то новое в когнитивную структуру и связать это с уже имеющимися знаниями, находящимися в этой структуре [31].

Современную реализацию концептуальных карт связывают с методиками британского психолога Тони Бьюзена (Tony Buzan), изобретателем интеллект-карт (Mind Maps) (также ментальные карты, диаграммы связей, карты разума) – популярной техники визуального мышления. Суть этой методики заключается в визуальном представлении информации, отражающем системные связи между целым и его частями. Такая диаграмма строится вокруг центральной идеи, концепции, темы или проблемы, от которой отходят «ветви» со связанными идеями. С помощью интеллект-карт можно структурировать любой материал – от простого списка литературы до учебного плана. В современных подходах к обучению интеллект-карты заслуженно занимают важное место, они помогают лучше воспринимать сложные концепции и надежнее запоминать изученное, делают обучение осмысленным, а не механическим. Также интеллект-карты являются эффективным инструментом планирования, помогают обрабатывать большие объемы информации, обеспечить всесторонний аналитический обзор по заданной теме, эффективно генерировать новые идеи, найти неочевидные решения и отказаться от непродуктивных вариантов [6,17,42].

В России примерно в это же время похожие идеи получили выражение в теоретических работах по системологии Г.П. Мельникова и П.Г. Кузнецова и широко использовались Г.П. Щедровицким и его последователями в организационно-деятельностных играх.

**Области применения.** Концептуальные карты успешно применяются в области образования как в качестве инструмента обучения, так и инструмента оценки успеваемости учащихся. Они способствуют осмысленному обучению, помогая создавать связи между тем, что уже знают учащиеся, и новыми знаниями, которые они получают. Концептуальные карты используются при решении множества задач обучения: структурирования материала для учебных курсов или программ, проведения эффективного мозгового штурма на определенную тему, проверки и оценки знаний учащихся.

Применение концептуальных карт в учебном процессе позволяет представить сложные понятия легко усваиваемым способом: структурировать и визуализировать сложные понятия, тем самым помогая учащимся увидеть связи между различными понятиями и идеями в предметной области. Использование концептуальных карт улучшает усвоение информации и запоминание материала, так как структурированное представление информации облегчает процесс объединения и ассоциации данных. Создание концептуальных карт учащимися во время изучения новой темы способствует стимуляции ассоциативного мышления, более глубокому пониманию материала, выявлению пробелов в знаниях и развитию

познавательных навыков. Также, концептуальные карты могут быть использованы для оценки знаний учащихся: проверки понимания материала и того, где их знания нуждаются в улучшении, выявления индивидуальных особенностей усвоения информации. Таким образом, использование концептуальных карт в образовании способствует улучшению качества обучения, развитию познавательных навыков и формированию глубокого понимания предмета, делая их ценным инструментом для педагогов и учащихся.

Помимо широкого распространения в сфере образования, концептуальные карты также часто используются в прикладных технических областях: инженерии, программировании, проект-менеджменте, бизнесе. Формализованные концептуальные карты используются при разработке программного обеспечения, при моделировании объектно-ролевых моделей, например, в формализме языка UML (Unified Modeling Language), а также как первый шаг при построении онтологий компьютерных систем. Концептуальные карты используются для графического представления учебных курсов, основанных на знаниях предметной области, как, например, в контекстно-зависимом планировщике путей обучения [27,46].

**Концептуальное картирование.** Метод визуализации взаимосвязей между различными понятиями, идеями, представлениями называется концептуальным картированием (Concept Mapping). Составление концептуальных карт позволяет глубоко рассмотреть предметную область, обнаружить неочевидные связи и генерировать нестандартные решения [2].

Стандартная процедура построения концептуальной карты включает в себя следующие этапы: определение центральной темы или основного вопроса; выявление и перечисление наиболее важных концептов, относящихся к данной теме; иерархическое упорядочивание концептов сверху вниз от самых абстрактных и общих до самых конкретных и специфических; а также добавление и маркировка связующих фраз. После создания предварительной концептуальной карты определяются и добавляются перекрестные ссылки, а затем выполняется проверка карты на полноту и корректность. Процесс расширения концептуальной карты может быть бесконечным, при необходимости можно добавлять новые концепты и взаимосвязи [43].

### 2.3. Онтологии

Онтология – одна из моделей представления знаний предметной области. Под онтологией предметной области традиционно понимается семантическое описание этой предметной области. Термины «онтологическое описание», «онтологическая модель», «онтология предметной области» используются как синонимы. Онтология описывает основные концепции (положения) предметной области и определяет отношения между ними [62,63].

В настоящее время для семантического моделирования преимущественно используется объектная онтология, описывающая текущее состояние некоторой предметной области как множества объектов, связанных друг с другом отношениями. Онтология вместе с набором индивидуальных экземпляров образует базу знаний (Knowledge Base).

Обычно онтологию представляют в виде ориентированного ациклического графа, вершинами которого являются объекты онтологии, а дугами – отношения между ними (свойства). Часто такую структуру из объектов и значений их свойств, построенную для определенной предметной области, называют графом знаний (Knowledge Graph) [4].

**Определение онтологии.** В настоящее время все еще не существует единого общепризнанного определения понятия «онтология», в научной литературе встречается множество различных определений, многие из которых противоречат друг другу.

Среди исследователей, занимающихся проблемами компьютерной лингвистики, наиболее традиционным (классическим) считается определение, данное в 1993 году Томом Грубером (Gruber T.A), американским учёным в области информатики, проделавшим фундаментальную работу по разработке онтологий. Онтология – это явная (формальная) спецификация концептуализации, которая имеет место в некотором контексте предметной области. Концептуализация представляет собой описание концепций (понятий), а также всю информацию, имеющую отношение к этим понятиям (свойства, отношения, ограничения, аксиомы, утверждения), необходимую для описания и решения задач в избранной предметной области. Формально онтология состоит из понятий (терминов, организованных в таксономию), их описаний и правил вывода [59-61,99].

Другое, часто используемое определение онтологии в информатике, представленное в Википедии, формулируется следующим образом: онтология предметной области – это попытка всеобъемлющей и детальной формализации некоторой области знаний с помощью концептуальной схемы (модели). Обычно такая модель состоит из иерархической структуры данных, содержащей все релевантные классы объектов, взаимосвязи и правила (ограничения), принятые в этой области. Концептуальная модель – система концептов (понятий) и отношений (связей) предметной области. Концепт – понятие, отражающее некоторый конкретный или абстрактный объект реального мира [25,55].

И еще одно из часто встречающихся в научной литературе определений онтологии: онтология – формальное явное описание понятий (или классов) в рассматриваемой предметной области, свойств (или слотов, ролей) каждого понятия, описывающих различные свойства и атрибуты понятия, и ограничений (или факетов, ограничений ролей), наложенных на слоты [15].

**Модель онтологии.** Классическая модель онтологии  $O$  формально определяется как упорядоченная тройка вида:  $O = \langle X, R, F \rangle$ , где  $X$  – конечное множество концептов (понятий, терминов) предметной области, которую определяет онтология  $O$ ;  $R$  – конечное множество отношений между концептами (понятиями, терминами) заданной предметной области;  $F$  – конечное множество функций интерпретации (аксиоматизация), заданных на концептах и/или отношениях онтологии  $O$ . На множества  $X$ ,  $R$ ,  $F$  накладываются следующие ограничения:  $X$  – непустое конечное множество ( $X \neq \emptyset$ ), а  $R$  и  $F$  – конечные множества.

**Свойства онтологии.** Множества  $R$  и  $F$  могут быть пустыми, что соответствует частным видам онтологии:

1. При  $R = \emptyset$  и  $F = \emptyset$  онтология трансформируется в простой словарь, или набор терминов, используемый в некоторой предметной области, без объяснений значений данных терминов. К примеру, простым словарем является любой орфографический словарь. Поскольку в таких простейших онтологиях явно не определяется смысл терминов, их использование ограничено. Представление онтологии в виде простого словаря достаточно эффективно лишь в тех случаях, когда определенные в онтологии термины принадлежат очень узкой области знаний, и их смысл уже заранее хорошо согласован в рамках определенного сообщества, например, индексы поисковых систем в сети интернет.

2. При  $R = \emptyset$  и  $F \neq \emptyset$  онтология преобразуется в глоссарий, или набор терминов с толкованием, в этом случае каждому элементу множества терминов из  $X$  ставится в соответствие функция (аксиома) и интерпретация (одна или несколько). К примеру, глоссарием является толковый словарь, позволяющий интерпретировать, уточнять и объяснять значения одних терминов на основе других, имеющихся в словаре.

3. Если  $R \neq \emptyset$  и  $F = \emptyset$  и множество отношений на концептах онтологии включает одно из отношений типа иерархии, этот случай соответствует специальному подклассу онтологий, называемыми простыми таксономиями. Под таксономической структурой понимается иерархическая система понятий, связанных между собой одним из отношений «класс-подкласс», «род-вид», «часть-целое» и т.п. При этом не приводится интерпретации понятий, смысл любого понятия раскрывается через указание его взаимосвязей с другими понятиями «вверх» и «вниз» относительно заданной таксономической структуры. Математически таксономией является древовидная структура классификаций определенного набора объектов. В отличие от онтологий, задача таксономий четко определена в рамках иерархической классификации объектов. Например, таксономиями являются используемые в библиографии классификационные системы, которые задают отношения иерархии между понятиями [28].

**Структура онтологии.** Большинство современных онтологий описывают экземпляры (индивиды), классы (понятия), атрибуты и отношения. В общем случае основными компонентами онтологий являются: экземпляры (или индивиды), классы (или понятия), атрибуты, отношения (или связи), аксиомы, правила и ограничения [26].

Экземпляры (instances) или индивиды (individuals) относятся к базовым компонентам самого нижнего уровня онтологии и представляют собой конкретные физические и абстрактные объекты. В литературе они также называются объектами, элементами, терминами. Экземпляры – это отдельные представители класса сущностей или явлений, то есть конкретные элементы какого-либо класса. Экземпляр может принадлежать одному или нескольким классам, или не принадлежать ни одному. Класс может иметь любое количество экземпляров. Главной целью онтологий является именно классификация экземпляров, и хотя их наличие в онтологии не обязательно, но, как правило, они присутствуют.

Классы (classes) или понятия (concepts) – это абстрактные наборы, коллекции или группы объектов (экземпляров). Также они называются множествами, коллекциями, категориями, типами. Классы могут включать в себя экземпляры, другие классы, либо сочетания и того, и другого. Классы или понятия используются в широком смысле, понятием может быть любая сущность, о которой может быть дана какая-либо информация. Понятия рассматриваются как концептуализации классов всех представителей некой сущности или явления. Каждый класс описывает группу индивидуальных сущностей, которые объединены на основании наличия общих свойств.

Класс может быть определен либо экстенциональным способом – указанием представителей класса, либо интенциональным способом – указанием условий. Соответственно, онтология подразделяется на экземпляры, которые являются объектами или событиями реального мира, и классы, которые являются наборами объектов реального мира. Определения класса задают свойства, которым должны соответствовать экземпляры, чтобы быть членами класса. А индивиды, которые удовлетворяют этим свойствам, называются экземплярами.

Классы (или понятия) являются общими категориями, которые могут быть упорядочены иерархически. Обычно классы организованы в таксономию, иерархическую классификацию понятий по отношению включения, при этом класс может быть подклассом другого класса, наследующим все его свойства. В качестве таксономического отношения используется отношение общее-частное, обладающее свойствами транзитивности и наследования.

Атрибуты (attributes) – аспекты, свойства, характеристики, параметры, особенности, которыми могут обладать объекты (а также классы и отношения). Наборы атрибутов

используются для хранения специфичной информации об объектах, которая характеризует объект и привязана к ним. Каждый атрибут имеет имя и значение, которое может быть сложным типом данных. Важная роль атрибутов состоит в том, чтобы определять отношения между объектами онтологии. Обычно отношением является атрибут, значением которого является другой объект. Онтологии являются истинными онтологиями только в том случае, если понятия связаны с другими понятиями (у понятий есть атрибуты).

Важная роль атрибутов заключается в том, чтобы определять отношения (зависимости) между объектами онтологии. Обычно отношением является атрибут, значением которого является другой объект.

Отношения или связи (relations) – способы, с помощью которых классы и экземпляры могут быть связаны друг с другом. Отношения также могут включать дополняющие и специфицирующие их атрибуты. Отношения между объектами в онтологии определяют, как объекты связаны с другими объектами. Как правило, отношение относится к определённому типу (или классу), который указывает, в каком смысле объект связан с другим объектом в онтологии. В совокупности набор отношений описывает семантику предметной области: то есть различные семантические отношения, такие как синонимия, гиперонимия и гипонимия, и т.п.

Базовыми в онтологии считаются отношения классификации (род/вид, класс/подкласс, специализация), которые ориентированы на построение иерархических таксономий, а также на логический вывод свойств объектов, в зависимости от отнесения их к тем или иным классам.

Самым распространенным типом отношений, используемым во всех онтологиях, является отношение категоризации, то есть отнесение к определенной категории (классу). Этот тип отношений имеет ряд других названий, встречающихся в литературе: таксономическое отношение, родовидовое отношение, класс – подкласс, гипоним – гипероним. Добавление в онтологию родовидовых отношений наследования вида «является подклассом» создаёт таксономию – древовидную структуру, наглядно показывающую, как объекты связаны друг с другом. Математически таксономия может быть представлена в виде дерева – связного ациклического графа. В такой структуре каждый объект является потомком родительского класса (в некоторых языках отношение «является подклассом» ограничено одним родителем для всех узлов, но во многих языках это не так).

Еще один распространённый тип отношений – мерологическое отношение включения «часть-целое», показывающее, как объекты объединяются в составные объекты. При включении в онтологию мерологических отношений возникающая иерархия больше не

может быть представлена в виде простой древовидной структуры, поскольку теперь элементы могут иметь более одного родителя. Таким образом, возникает новая структура, представляющая собой ориентированный ациклический граф.

Аксиомы, правила и ограничения выражают суждения, которые имеют место в рассматриваемой предметной области.

Аксиомы (axioms) – утверждения (включая правила) в логической форме, которые вместе составляют общую теорию, которую онтология описывает в своей области применения. Моделируют предложения, которые всегда истинны.

Правила (rules) – утверждения в форме предложения «если–то» (предшествующее – последующее), описывающие логические выводы, которые могут быть сделаны из утверждений.

Ограничения (restrictions) – формально сформулированные описания того, что должно выполняться, чтобы какое-то утверждение было принято в качестве входных данных.

Аксиомы задают условия соотнесения классов и отношений, они выражают очевидные утверждения, связывающие классы и отношения. Под аксиомой можно понимать утверждение, вводимое в онтологию в готовом виде, из которого могут быть выведены другие утверждения. Аксиомы позволяют выразить ту информацию, которая не может быть отражена в онтологии посредством построения иерархии классов и установления различных отношений между ними. Аксиомы позволяют в дальнейшем выстраивать логические утверждения в рамках онтологии. Они могут снабжать исследователей информацией о правилах, позволяющих автоматически добавлять информацию. Аксиомы могут также представлять собой ограничения, накладываемые на какие-либо отношения, делающие возможным выведение умозаключений.

Компоненты онтологии образуют определенную иерархию, на нижнем уровне которой находятся экземпляры, далее идут классы, выше располагаются отношения между классами, а обобщающими и связующими являются аксиомы [16].

## 2.4. Байесовская сеть

Один из самых сложных методов, применяемых для реализации адаптации учебных курсов в интеллектуальных системах обучения – байесовская сеть, которая используется для прогнозирования вероятности того, что учащиеся овладеют новыми знаниями, основываясь на характеристиках, установленных перед началом процесса обучения [29].

Байесовская сеть (Bayesian network) – это вероятностная графическая модель, которая представляет набор переменных и их условные зависимости с помощью ориентированного

ациклического графа. Его вершины – некоторые переменные, а ребра – причинные отношения между переменными. Каждая вершина имеет ассоциированную таблицу условных вероятностей, которые зависят от характеристик учащегося [79].

Вероятность изменяется соответственно с прогрессом успеваемости учащегося, посредством выбора и адаптации метода обучения. Самой трудной задачей при использовании байесовской модели является определение априорных вероятностей, основанных на предварительном тестировании [91].

Из рассматриваемых в статье интеллектуальных систем обучения, байесовская вероятностная модель обучаемого (Bayesian probabilistic student model) используется в системе AST (Adaptive Statistics Tutor), веб-ориентированной обучающей системе по статистике, и системе KBS Hyperbook, адаптивной гиперкниге для курса по информатике и вычислительной технике [97,65].

### **3. Обзор адаптивных систем обучения с генерированием электронных учебных курсов**

Впервые концепция адаптивных электронных учебных курсов была предложена в 1990 году в системе GTE (Generic Tutoring Environment), программной среде для разработки интеллектуальных учебных курсов, основанных на общих знаниях, включающих задачи, методы и объекты [100-102].

Год спустя была разработана система ECSAI (Environnement de Conception de Systeme d'Apprentissage) – генератор обучающей среды с интеллектуальным упорядочением учебных модулей [56-58].

Динамический генератор учебных курсов DCG (Dynamic Courseware Generator) – не первая, но одна из самых примечательных систем. В DCG дифференцируются концепты предметной области, представленные в виде графа (планирование контента) и учебные материалы (планирование представления) [103-105].

Наиболее цитируемая в литературе система ELM-ART (Episodic Student Modelling Adaptive Remote Tutor) представляет собой интеллектуальную веб-среду для изучения LISP. Эта система была разработана в 1996 году на основе ELM-PE (среды программирования ELM), которая поддерживает программирование на основе примеров, интеллектуальный анализ решения задач, расширенное тестирование и отладку. В обеих системах применяется эпизодическое моделирование учащихся. Выводы о знаниях учащихся основываются на просмотренных или прочитанных ими страницах [40,109].

В 1997 году была создана система AST (Adaptive Statistics Tutor) – веб-ориентированный электронный учебный курс для изучения статистики [97].

Через два года после появления ELM-ART и через шесть лет после разработки DCG, в 1998 году, появилась среда для разработки адаптивных учебных курсов ACE (Adaptive Courseware Environment), которая объединила адаптивную навигационную поддержку из ELM-ART с планированием представления из DCG [96].

Последней системой XX века стала система KBS Hyperbook, адаптивная гиперкнига для учебного курса по информатике и вычислительной технике [65].

Первое десятилетие XXI века было очень продуктивным, особенно первые два года. В 2001 году была создана система ATLAS (Authoring Tool for Adaptive Software Design), графический инструмент для разработки интерактивных динамических адаптивных учебных курсов. ATLAS давал представление о взаимосвязях между структурой, содержанием и представлением учебных материалов и моделью обучаемого. Система предоставляла преподавателю возможность создания и редактирования учебного материала и назначения заданий. Для создания нового учебного курса требовалось сделать схематическое разбиение заданий и установить между ними продукционные связи [77,80].

Также в 2001 году появилась адаптивная гипермедиа-система под названием INSPIRE (Intelligent System for Personalized Instruction in a Remote Environment) [90].

Еще одна система, обязанная своим происхождением системе ELM-ART – система WINDS (Web-based Intelligent Design and Tutoring System), программное средство для управления контентом, тесно связанная с ACE. Она была разработана для создания индивидуализированных учебных курсов [95].

Система NetCoach, созданная в 2002 году, как и система ACE, основанная на ELM-ART, представляла собой среду для разработки и создания веб-ориентированных адаптивных учебных курсов [108].

В 2002 году была разработана система APeLS (Adaptive Personalized e-Learning Service), первая система, которая представляла учебные курсы с использованием кандидатных групп и описаний. Архитектура системы включала модель контента, модель обучаемого и модель описаний [48].

В 2004 году, основываясь на идеях APeLS, была разработана система ACCT (Adaptive Course Construction Toolkit), предоставляющая набор инструментов для создания как адаптивных, так и традиционных электронных учебных курсов. ACCT включала средства для генерации онтологий предметной области, описаний, тестов и баз данных учебных

материалов, а также поддерживала экспорт учебных программ в APeLS вместе со знаниями и описаниями предметной области [49,50].

В 2005 году в системе ASM (Adaptive Sequencing Methodology) была предложена методология адаптивного упорядочения, в которой были использованы статистические методы поиска наилучшей программы обучения [68].

В 2006 году в системе iClass (Intelligent Distributed Cognitive-based Open Learning System for Schools) были продолжены исследования, проводимые в рамках систем APeLS и ACCT. iClass генерировала электронные учебные курсы, адаптирующиеся в соответствии с когнитивными характеристиками учащихся. В отличие от APeLS, iClass разделяла педагогическую информацию и знания предметной области [87].

В 2009 году в системе PAIGOS был реализован новый метод генерирования электронных учебных курсов, основанный на модели, обладающей важными педагогическими знаниями и не зависящей от теории обучения [98].

Создание учебных курсов в PAIGOS было основано на планировании иерархической сети задач (HTN планировании), которое было впервые предложено в среде дистанционного обучения FORHUM (Formación Humana) [78].

ActiveMath – интеллектуальный генератор электронных учебных курсов, использующийся в PAIGOS [80].

Один из самых современных подходов применяется в системе ADOPTA (ADaptive technOlogy-enhanced Platform for eduTAinment), программной платформе для генерирования адаптивных электронных учебных курсов в интеллектуальных системах обучения, появившейся в 2009 году. Учебные курсы в этой системе названы «раскадровками» («storyboard») [36].

Концепция полноценной персонализации сценариев электронного обучения была реализована в системе ELP1+ELP2 (ELearning personalization levels 1 and 2). Основная цель системы – позволить преподавателям выбирать и применять стратегию персонализации, соответствующую характеристикам учащихся и специфике учебного курса. Первый уровень персонализации ELP1 позволяет персонализировать учебные материалы и структуру учебного курса в соответствии с данной (специфицированной в ELP2) стратегией персонализации (примененной к выбранному сценарию обучения). Второй уровень персонализации ELP2 позволяет определить стратегию персонализации при помощи выбора подмножества параметров персонализации [53].

Разработанная в 2012 году информационная система ISCARE (Information System for Competition based on pRobleM solving in Education) – интеллектуальная система обучения,

основанная на швейцарской системе проведения турниров, которая предоставляет учащимся возможность соревноваться, чтобы улучшить процесс обучения. Соревнование основано на различных турнирах и раундах. В каждом раунде учащиеся разбиваются на группы по два человека, соревнующихся друг с другом, и каждая пара получает разные задачи, которые требуется решить за ограниченный период времени [83].

UZWEBMAT – среда электронного обучения для изучения теории вероятности из курса математики, использует экспертную систему для представления учебного материала в соответствии со стилем обучения учащихся и адаптации материала в соответствии со знаниями учащихся [89].

Oscar CITS (Oscar Conversational ITS) – современная интеллектуальная система, созданная в 2014 году, использующая естественно-языковой интерфейс и предоставляющая учащимся возможность формирования знаний посредством дискуссии. Oscar CITS ставит своей целью моделировать индивидуальное обучение с преподавателем посредством динамического выявления стиля обучения учащихся и последующей адаптации к нему, настраивая диалоговый учебный курс. Oscar CITS не зависит от конкретной модели стилей обучения и изучаемой предметной области [76].

В 2015 году была создана система TECH8 – интеллектуальная адаптивная система электронного обучения, имеющая модульную структуру и основанная на системе сбора метаданных и переменных, имеющих важное значение для процесса обучения. Система, предварительно подготовленная таким образом, поддерживает индивидуализацию и дифференциацию и может быть адаптирована к уровню знаний и пониманием предмета каждым учащимся [52].

### **3.1. Формализация знаний предметной области и модели обучаемого в адаптивных системах обучения**

Первое время знания предметной области в адаптивных системах обучения были представлены в форме графов или сетей. Альтернативные подходы различались видом представления вершин в этих структурах.

В системах ECSAI и ELM-ART знания предметной области организованы в иерархическую сеть концептов. В некоторых системах (DCG, ACE, INSPIRE) для представления знаний предметной области используются графы концептов, при этом каждый концепт связан с некоторым набором учебных материалов (текст, пример, демонстрация, задача, тест). В этой самой простой модели предметной области представлен только один

тип связи – предварительное условие (посылка) (prerequisite), что обеспечивает структурирование контента [41].

В NetCoach знания предметной области представлены в форме концептуальной сети, при этом концепты представляют страницы, содержащие учебный материал, который нужно изложить учащемуся. Есть два типа связей в этой концептуальной сети: предварительное условие (посылка) (prerequisite) и заключение (вывод) (conclusion) [108].

Система KBS Hyperbook использует графы зависимостей, вершины которых – концепты (элементы знаний), а ребра – взаимосвязи между концептами [65].

С 2004 года к формализации знаний предметной области стал применяться онтологический подход. В системах APeLS, ACCT, ASM, iClass и ADOPTA знания предметной области выражены онтологией.

APeLS и ACCT представляют знания предметной области в виде концептуальной онтологии, представляющей собой коллекцию описаний концептов верхнего уровня, связей и взаимосвязей, независимым от содержания способом [48,49].

Граф путей концептов в ASM представляет концептуальную структуру онтологии знаний предметной области. Концепты выбираются из графа путей концептов согласно взаимосвязям между иерархией учебных целей и онтологией предметной области [67].

В ADOPTA все обучающие материалы подразделяются на учебные объекты, которые могут быть описаниями, заданиями или вопросами. Учебные объекты организованы в онтологический семантический граф с двумя видами связей: «является» (is\_a) для связей между описаниями и «имеет» (has\_a) для связей между заданиями, вопросами и описаниями [36].

Что касается подходов к моделированию учащихся, в большинстве рассматриваемых в статье систем применяется оверлейная модель. В этой модели знания учащегося фактически являются взвешенным подмножеством экспертных знаний [45].

Система ECSAI использует простую оверлейную модель, без весов. В ELM-ART, для каждой посещенной страницы, соответствующие элементы в оверлейной модели учащегося помечаются как посещенные. После теста или решения задачи, все изученные концепты из модулей помечаются как известные, и начинается процесс формирования выводов относительно знаний учащегося [40,56].

ASM использует оверлейную модель для определения уровня знаний учащихся и отслеживает число баллов, набранных в тестах, а также число попыток решения тестов. GTE использует взвешенную оверлейную модель. Модель обучаемого в DCG – численная

оверлейная модель с концептуальной структурой, при этом знание учащимся каждого концепта представляется числом внутри заданного интервала [68,100,105].

NetCoach использует многоуровневую оверлейную модель. Первый уровень показывает, посещал ли учащийся страницу, второй уровень содержит информацию об ответах на вопросы, третий уровень показывает, были ли получены сведения о том, что учащийся знает концепт, основываясь на связи между этим концептом и концептами, которые учащимся уже изучены. Четвертый уровень показывает, отметил ли учащийся некоторый концепт как известный [108].

В системе ASM применяется стереотипная модель, которая используется для представления стилей обучения в соответствии с моделью Хани-Мамфорда. Эта стереотипная модель также используется для способа представления учебного материала (визуальный, текстовый, аудиальный и смешанный) [7,11,66].

Более сложные подходы к моделированию учащихся позволяют реализовать более сложную адаптацию. В системах AST и KBS используется байесовская вероятностная модель обучаемого (Bayesian probabilistic student model), а в системах ACE и WINDS – эпизодическая модель обучаемого (episodic student model) [95-97,107].

В KBS Hyperbook применяется байесовская вероятностная модель обучаемого, где узлы – концепты. Зависимости между концептами представлены условными вероятностями. В AST каждое взаимодействие с системой оказывает влияние на модель обучаемого. Влияние зависит от образовательного контента и его параметров. В зависимости от сложности и значимости, показатель надежности в Байесовской вероятностной модели обучаемого увеличивается [65,97].

Эпизодическая модель обучаемого в WINDS хранит все учебные состояния (события). Эти состояния называются «эпизодами» и включают действия учащегося, оценки его работы преподавателем и системой, а также любые выводы системы об уровне знаний учащегося [95].

Модель обучаемого в ACE комбинирует оверлейную и эпизодическую модель, она состоит из трех частей: профиля, модели знаний и модели интересов [96].

Модель обучаемого в APeLS содержит априорные знания и цели обучения. Модель обучаемого в ADOPTA включает цели и предпочтения, стиль обучения и знания учащегося [36,48].

### **3.2. Генерирование и адаптация электронного учебного курса**

Процесс генерирования электронного учебного курса включает в себя следующие три этапа: генерирование элементов учебного курса, выбор и упорядочение элементов учебного курса, представление элементов учебного курса [8].

### 3.2.1. Генерирование элементов электронного учебного курса

Первый этап – это генерирование элементов электронного учебного курса. Он может быть выполнен или самим преподавателем, или же элементы могут быть сгенерированы автоматически. В большинстве рассматриваемых в статье систем элементы учебного курса генерируются преподавателем (GTE, ECSAI, AST, KBS Hyperbook, ATLAS, INSPIRE, WINDS, NetCoach, APeLS, ACCT, ADOPTA, ELP1+ELP2, IS CARE, UZWEBMAT, Oscar CITS, TECH8).

Элементы учебного курса в GTE and ATLAS – задания, или ряд действий, которые требуется выполнить в течение процесса обучения. В GTE задания могут быть следующими: «привести пример», «привести противоположный пример», «выполнить упражнение», «сделать обзор» [100].

Элементы учебного курса в ECSAI – учебные модули, являющиеся совокупностью содержательных, практических и оценочных элементов, которые скомбинированы для достижения единой цели обучения. Каждый учебный модуль связан с описанием, которое включает: метку, тип (представление, примеры, упражнения), текстовое описание, предусловия (завершенные уроки, понимание учащимся некоторых элементов знаний предметной области) и постусловия (изменения в модели обучаемого после завершения модуля) [58].

В AST существуют различные типы элементов учебного курса – уроки, разделы, подразделы и концепты. При этом каждый элемент учебного курса имеет так называемые предпосылки (учебные модули, которые учащийся должен знать) и последствия (возможное влияние на другие учебные модули) [97].

Учебный курс в KBS Hyperbook состоит из модулей, которые соответствуют частям книги и семантически связаны, при этом каждый модуль индексирован концептами предметной области [65].

В INSPIRE элементы учебного курса – модули, которые включают цели обучения, итоговые концепты и учебный материал, связанный с итоговыми концептами и их предварительными концептами. Концепты, являющиеся самыми важными для достижения цели, называются итоговыми концептами. С каждым итоговым концептом связаны предварительные и зависимые концепты [90].

В APeLS учебный курс подразделяется на секции, модули и элементы. Курс – это последовательность шагов по контенту, каждый шаг состоит из набора учебных материалов, имеющих одну и ту же учебную цель, т.е. принадлежащих одной «кандидатной группе» (candidate group). Каждая «кандидатная группа» содержит учебные объекты, отвечающие одним и тем же требованиям к контенту. Учебные объекты в «кандидатной группе» могут отличаться друг от друга техническими, образовательными или любыми другими характеристиками, которые могут быть адаптированы. Разделение контента и его характеристик способствует повторному использованию учебных объектов [48].

В ACCT элементы учебного курса – концептуальные описания (concept narratives). Описательная модель содержит в себе семантику применяемой педагогической стратегии. Она описывает логику выбора и представления учебных действий и концептов. Концептуальное описание позволяет преподавателю применять аспекты педагогической стратегии к некоторым частям адаптивного учебного курса [49].

В ADOPTA элементы курса – учебные объекты, которые преподаватель помещает на страницы так называемой «раскадровки» (storyboard) учебного курса. Для каждого учебного объекта преподаватель задает коэффициент уровня знаний, который используется механизмом адаптации для адаптивного выбора контента [36].

В UZWEBMAT во время подготовки элементов курса (учебных объектов) особое внимание уделяется тому, чтобы учебные объекты учитывали особенности каждого стиля обучения. Например, диаграммы, блок-схемы, рисунки и анимации подходят для учащихся визуального типа, а звуковые инструкции, оповещения и обратная связь – для аудиального типа [89].

В ELM-ART элементы учебного курса – модули, иерархически организованные в уроки, разделы, подразделы и конечные страницы [40].

Структура курса в ACE – концептуальная сеть из модулей, которые могут быть секциями (могут включать другие модули) или концептами. Каждый учебный модуль включает т.н. «предварительные» модули, которые учащиеся должны знать перед изучением определенного модуля. Предварительные модули имеют веса в соответствии со своей значимостью [96].

В DCG элементы учебного курса – HTML страницы, связанные с концептами предметной области. Генерирование учебного курса учитывает текущие знания учащихся и принимает во внимание различия в способе и скорости приобретения знаний [105].

В PAIGOS элементы учебного курса – структурированные наборы ссылок на учебные материалы [98].

Учебный курс в системе TECH8 состоит из шагов обучения и оценивания. Каждый шаг обучения имеет разветвленную структуру с многочисленными формирующими оценками [52].

### 3.2.2. Выбор и упорядочение элементов учебного курса

Второй этап генерирования учебного курса – это выбор и упорядочение элементов курса. Этот процесс может быть статическим (выполняться только один раз перед началом обучения) или динамическим (повторяться несколько раз в течение процесса обучения).

В большинстве рассматриваемых в статье систем (ECSAI, DCG, ELM-ART, AST, ACE, KBS Hyperbook, ATLAS, INSPIRE, NetCoach, APeLS, ACCT, ASM, ADOPTA, ISCARE, UZWEBMAT, TECH8) используется динамический выбор и упорядочение элементов курса.

В DCG используется планировщик контента, который генерирует пути в графе концептов предметной области. Эти пути служат шаблоном для планирования представления. Впоследствии планировщик представления выбирает учебный материал, связанный с выбранными концептами предметной области, и определяет порядок и способ, которыми этот материал будет представлен учащемуся [105].

В ELM-ART каждый модуль содержит статические слоты для текста, который будет показан, и для информации, которая будет использоваться для связывания модулей с концептами. Модули также могут иметь динамические слоты для тестов, содержащих вопросы, на которые учащийся должен ответить, дополнительно к описанию задачи. В процессе формирования выводов об учащемся все концепты, которые служили предпосылками для модуля, считаются изученными. Информация из динамических слотов используется для автоматической генерации оптимального пути (плана) обучения для учащегося [40].

В AST имеется стратегия обучения по умолчанию для каждого типа концептов (обучение на примере, обучение с помощью чтения текста, обучение на собственном опыте), хотя преподаватель может изменять ее по мере необходимости. Кроме того, для каждой стратегии обучения есть правила, допускающие адаптивный выбор стратегии обучения в зависимости от характеристик учащегося и типа изучаемых концептов. Система отслеживает, какую комбинацию и порядок учебных материалов учащийся использует чаще всего, и соответствующим образом изменяет стратегию обучения [97].

AST определяет, какие модули учащийся должен изучать далее, базируясь на байесовской вероятностной оверлейной модели и требованиях для возможных следующих модулей. Система сначала отбрасывает те модули, для которых учащийся не выполнил необходимых

условий, а затем обращает внимание на достоверность, надежность и вес модулей, для которых учащийся выполнил необходимые условия. Следующим модулем выбирается тот, для которого учащийся достоверно выполнил самые важные условия.

В ACE преподаватель должен явно определить, какие стратегии обучения будут использоваться для каждого типа концептов. Компонент адаптивного упорядочения системы пытается придерживаться пути обучения, определенного текущим знанием учащегося. Кроме того, упорядочение адаптируется к интересам, установленным учащимся [96].

В KBS Hyperbook учащиеся выбирают цель обучения, и система предлагает проектные модули, с которыми они должны поработать для достижения своей цели (адаптивный выбор проекта). Кроме того, система может предложить цели обучения, которые совместимы со знаниями учащегося (адаптивный выбор цели): система предлагает проекты и затем генерирует учебный курс [65].

На основании цели обучения, выбранной учащимся, INSPIRE генерирует элементы учебного курса (уроки), совместимые со стилем обучения и знанием учащегося [90].

В NetCoach цель обучения – множество концептов, которые учащиеся должны изучить. Все необходимые условия автоматически определяются и предлагаются подходящие элементы учебного курса (страницы). На основании описаний концептов все элементы курса генерируются индивидуально в соответствии с моделью учащегося [108].

План обучения в iClass состоит из элементов учебного курса (концептов), которые определены согласно выбранной стратегии обучения [87].

### **3.2.3. Представление элементов электронного учебного курса**

Представление элементов электронного учебного курса – третий этап генерирования учебного курса, где также применяется адаптация. Большинство рассматриваемых в статье систем используют методы адаптивной гипермедиа, такие как адаптивная поддержка навигации и адаптивное аннотирование гиперссылок (ECSAI, ELM-ART, AST, ACE, KBS Hyperbook, INSPIRE, WINDS, NetCoach, ADOPTA) [7,11,13,38].

ELM-ART использует технику адаптивного аннотирования на основе расширенной «модели светофора». В этой модели зеленая точка перед гиперссылкой указывает на рекомендуемую для изучения информацию, красная точка показывает, что у учащегося недостаточно знаний для понимания данной информации, а желтая точка означает, что эта гиперссылка не ведет к новой информации для учащегося. Помимо этого, в системе существует возможность следовать оптимальному плану обучения, который определяется текущим знанием учащегося. Следующая страница в оптимальном плане обучения

соответствует странице, помеченной как «рекомендуемая». Подобный метод используется в системе AST [40].

KBS Hyperbook также применяет адаптивную навигацию для адаптации списков модулей учебного курса [65].

ACE реализует два метода поддержки адаптивной навигации: адаптивное аннотирование и пошаговое связывание гиперссылок. Адаптивное аннотирование гиперссылок предоставляет учащемуся дополнительную информацию относительно контента по гиперссылке (используя различные цвета). Аннотирование адаптируется к учащемуся, принимая во внимание его знания и связи между модулями, которые должны быть изучены (уже просмотренные концепты; концепты, для которых у учащегося не хватает предварительных знаний; предпочтительные концепты и модули, не рекомендуемые, но не требующие дополнительных знаний) [96].

В WINDS структура учебного курса и модель обучаемого допускают адаптивную навигацию и представление. Поскольку учебные объекты скомпонованы, могут быть использованы различные техники адаптации: прямое руководство, сортировка, сокрытие и аннотирование гиперссылок. Учебные элементы состоят из контент-блоков, которые имеют предопределенную последовательность и учебную роль, что позволяет приложениям использовать различные методы адаптации: дополнительные, предварительные и сравнительные пояснения, варианты пояснений и сортировку гиперссылок [95].

NetCoach использует адаптивное аннотирование гиперссылок. Следующая страница, рекомендуемая учащемуся, динамически генерируется на основании целей обучения и знаний учащегося. Учащиеся получают предупреждение, если они выбирают страницу, для которой у них нет предварительных знаний (предупреждения можно отключить) [108].

Для представления учебного курса в ADOPTA используются адаптивная навигация, аннотирование и выбор контента [36].

В системах GTE, DCG, APeLS и ASM для представления элементов учебного курса используется подход, отличный от адаптивной гипермедиа. В GTE для адаптации применяется динамический выбор методов обучения. Перед выполнением определенной задачи GTE выбирает наиболее подходящий метод обучения, связанный с этой задачей [102].

DCG использует план представления для каждого концепта. Во время представления учебного курса, если учащиеся правильно отвечают на вопросы теста, то они продолжают процесс обучения в соответствии с определенным планом, который не меняется. Если учащиеся демонстрируют недостаток знаний, учебный курс перепланируется. Перепланирование осуществляется или на уровне представления, посредством изменения

последовательности учебного материала, или посредством изменения методов обучения для представления концепта. Если учащиеся снова демонстрируют недостаток знания, планировщик контента генерирует новое подмножество концептов, ведущих к пониманию заданного концепта в соответствии с текущим знанием учащегося [105].

В APeLS планирование представления ограничено выбором кандидатной группы. Кандидатная группа предопределяется множеством различных структур и форматов для учебных материалов [48].

В ASM граф планов обучения строится путем замещения каждого концепта в графе путей концептов на соответствующую последовательность учебных материалов [68].

#### **3.2.4. Функции адаптации**

Следующий четырехфазный циклический процесс адаптации электронных учебных курсов в системах обучения был предложен в 2008 году [93]:

- 1) сбор информации об обучаемом;
- 2) создание и поддержание модели обучаемого;
- 3) выбор элементов учебного курса на основании модели обучаемого;
- 4) представление элементов учебного курса на основании модели обучаемого.

Существует много различных классификаций когнитивных стилей и стилей обучения, также как и противоречивых мнений о том, какой подход наилучший. Очевидно, что учащиеся отдают предпочтение одному из способов обучения, но неясно, насколько стабильны эти предпочтения, то есть изменяются ли они в зависимости от контента или настроения учащегося, и насколько они надежны для определения способа, которым учащийся предпочитает обучаться. Информация об учащемся, полученная посредством проверки его знаний, более надежна, чем выводы, сделанные на основании истории его навигации [7,11,77].

Проблемы измерения и оценки уровня знаний, умений и навыков учащихся в настоящее время являются одними из центральных в практике электронного обучения. Если цель обучения определяет, что должны знать и уметь учащиеся, то задачи обучения отвечают на вопрос, как двигаться к этой цели. В 1956 году американским психологом методов обучения Бенджамином Блумом (Benjamin Samuel Bloom) была создана первая таксономия целей обучения. Она является самой известной моделью, описывающей процесс мышления, и включает шесть навыков мышления, структурированных от самого базового до самого продвинутого уровня. С точки зрения Блума, цели обучения напрямую зависят от иерархии мыслительных процессов, их еще называют элементами таксономии Блума: знание

(knowledge), понимание (comprehension), применение (application), анализ (analysis), синтез (synthesis) и оценка (evaluation). Таксономия Блума помогает структурировать учебный процесс и проверять глубину знаний обучающихся, обеспечивая четкую и стабильную базу для разработки адаптивных систем обучения [7,11,35].

В рассматриваемых в статье системах основные механизмы адаптации основаны на стилях обучения (AST, ATLAS, INSPIRE, WINDS, APeLS, ACCT, ASM, ADOPTA, ELP1+ELP2, UZWEBMAT, Oscar CITS) и целях обучения (KBS Hyperbook, NetCoach, ACCT, iClass). Несколько систем адаптируют учебный курс в соответствии с предпочтениями или интересами учащихся (DCG, ACE, WINDS, iClass), когнитивным стилем (DCG) или поведением учащихся (ECSAI).

Подавляющее большинство систем применяет адаптацию в соответствии со знанием учащихся. Этот вид адаптации не влияет на изменение содержания элементов учебного курса, он скорее заключается в выборе и упорядочении элементов учебного курса (ECSAI, DCG, ELM-ART, AST, ACE, KBS Hyperbook, INSPIRE, WINDS, NetCoach, APeLS, ACCT, ASM, iClass, ADOPTA, ELP1+ELP2, ISCARE).

Адаптация электронного учебного курса может быть достигнута или выбором и упорядочением элементов курса, или адаптацией представления элементов курса. При адаптации одним из ключевых моментов является использование как можно более мелких элементов учебного курса. Чем глубже уровень, тем адаптация точнее и чувствительнее, и, в конечном счете, тем она успешнее.

Рассматриваемые в статье системы в большинстве своем в качестве уровней адаптации используют модули (units) (ECSAI, ELM-ART, AST, ACE, KBS Hyperbook, ELP1+ELP2), страницы (pages) (NetCoach), уроки (lessons) (INSPIRE), задачи (tasks) (ATLAS, ISCARE), темы (themes) (GTE), учебные объекты (learning objects) (WINDS, iClass, ADOPTA, UZWEBMAT, Oscar CITS) или кандидатные группы (candidate groups) (APeLS, ACCT). Только две системы (DCG, ASM) используют наименьшие элементы, допустимые в качестве уровня адаптации – концепты предметной области.

### **3.3. Тестирование знаний**

Тестирование знаний учащихся – одна из важных функций, предоставляемых системами обучения. В большинстве рассматриваемых в статье систем преподаватель должен сам формулировать вопросы для проверки знаний учащихся. Недостаток такого подхода в том, что преподавателям вручную требуется вводить вопросы, определять методы оценивания, выбирать правильные и неправильные ответы и выставять оценки.

При исследовании задачи тестирования знаний в адаптивных системах важно различать, генерирует ли система вопросы сама (автоматически), или же преподаватель вводит вопросы (вручную). Во многих системах тестирование знаний не проводится вообще (GTE, ECSAI, KBS Hyperbook, ATLAS, APeLS, ACCT, ELP1+ELP2, ISCARE, Oscar CITS). В некоторых системах вопросы формулируются преподавателем (DCG, ELM-ART, AST, ACE, INSPIRE, WINDS, NetCoach, ADOPTA, UZWEBMAT, TECH8). Для систем ASM, iClass и PAIGOS, тестирование знаний упоминается, но явно не описывается.

При этом нет ни одной системы, в которой вопросы для тестирования полностью автоматически генерируются, основываясь на модели предметной области. Только в системе ACE проводится предварительный тест знаний учащихся, который генерируется динамически и включает все концепты первого уровня и связанные с ними тесты. Во время прохождения учебного курса знания учащихся проверяются, и система динамически адаптируется к изменяющимся знаниям. В системе DCG учащиеся могут сами по желанию пройти тесты на знание любого концепта в любое время. Упражнения и тесты представлены в виде небольших модулей, содержащих сохраненные правильные ответы, объяснения и т.д.

#### **4. Заключение**

В наш век стремительного развития информационно-образовательных ресурсов, сетевых дистанционных технологий и внедрения электронного обучения в современную систему образования разработка адаптивных электронных учебных курсов, предоставляющих индивидуализированный учебный материал для каждого учащегося и динамически выбирающих оптимальный метод обучения на каждом шаге процесса обучения, является актуальным и перспективным направлением исследований.

В статье дан обзор интеллектуальных систем обучения с адаптивным генерированием электронных учебных курсов, приведены методы и средства адаптации, применяемые на этапах генерирования, выбора и упорядочения, и представления элементов учебного курса. Рассмотрены вопросы формализации предметной области и модели обучаемого, функции адаптации и тестирования знаний учащихся в данных системах. В рамках статьи представлены методы, применяемые для реализации адаптации электронных учебных курсов: графы путей концептов, графы планов обучения, графы учебной деятельности, концептуальные карты, онтологии, байесовские сети.

Рассмотрев и проанализировав преимущества и недостатки методов и средств, используемых для адаптации в описанных в статье системах обучения, можно прийти к выводу, что модель адаптивного генерирования электронных учебных курсов должна

удовлетворять следующим требованиям: генерирование элементов учебного курса должно быть автоматическим и основанным на структуре знаний предметной области; выбор и упорядочение элементов учебного курса должны быть динамическими и базироваться на знаниях обучаемого; вопросы и тесты для проверки знаний учащихся должны автоматически генерироваться, исходя из структуры знаний предметной области.

Также в модели, удовлетворяющей вышеперечисленным требованиям, должен применяться онтологический подход к формализации знаний предметной области; модель обучаемого должна комбинировать оверлейную модель с байесовской вероятностной моделью. Преподаватель не должен влиять на генерирование элементов учебного курса, этот процесс должен быть полностью автоматизирован, базируясь на онтологии предметной области. Уровень знаний учащихся должен определяться согласно таксономии Блума. Адаптация электронного учебного курса может быть достигнута выбором и упорядочением элементов курса или адаптацией представления элементов курса. Поскольку при адаптации одним из ключевых моментов является использование насколько возможно более мелких элементов учебного курса, в качестве уровня адаптации должны использоваться концепты знаний предметной области, как самые наименьшие и неделимые частицы знаний.

Тестирование знаний учащихся должно быть автоматизировано и исключать любое вмешательство преподавателя, вопросы должны генерироваться автоматически и быть связанными с тестами, которые тоже должны оцениваться автоматически. Вопросы тестов должны генерироваться, основываясь на концептах и отношениях между ними в онтологии предметной области с использованием шаблонов вопросов. Уровень сложности вопросов также должен соответствовать уровню знаний учащихся, используя таксономию Блума.

## Список литературы

1. Абасов З.А. Индивидуализация обучения - путь развития индивидуальности ученика // Народное образование. 2015. № 4. С. 134-142.
2. Абрамова Н.В., Николаева Ю.В. Построение концептуальных карт как метод повышения валидности результатов оценочного исследования // Социология: методология, методы, математическое моделирование. 2006. № 23. С. 83-99.
3. Баранников К., Ананин Д., Стрикун Н., Алканова О., Байзаров А. Гибридное обучение: российская и зарубежная практика // Вопросы образования. 2023. №2. С. 33-69.
4. Болдачёв А.В. Событийная онтология vs объектная. 21.12.2022. [Электронный ресурс]. URL: <https://habr.com/ru/post/706916/>
5. Бурняшов Б.А. Персонализация как мировой тренд электронного обучения в учреждениях высшего образования // Современные проблемы науки и образования. 2017. № 1.

6. Бьюзен Т. Интеллект-карты. Полное руководство по мощному инструменту мышления / пер. с англ. Ю. Константиновой. М.: Манн, Иванов и Фербер, 2019. 208 с.
7. Волянская Т.А. Адаптивное генерирование учебных курсов в интеллектуальных системах дистанционного обучения. Часть 1. Новосибирск, 2019. 54 с. (Препринт / ИСИ СО РАН; № 183).
8. Волянская Т.А. Адаптивное генерирование учебных курсов в интеллектуальных системах дистанционного обучения. Часть 2. Новосибирск, 2019. 40 с. (Препринт / ИСИ СО РАН; № 184).
9. Волянская Т.А. Виртуальный музей истории информатики в Сибири // Материалы Междунар. конф. молодых ученых по математическому моделированию и информационным технологиям. Новосибирск, 2002. С. 49.
10. Волянская Т.А. Виртуальный музей истории информатики в Сибири: модель предметной области и модель пользователя // Новые информационные технологии в науке и образовании. Новосибирск, 2003. С. 124-146.
11. Волянская Т.А. Вопросы адаптивности в системах дистанционного обучения // Системная информатика. 2020. № 16. С. 11-46.
12. Волянская Т.А. Методы адаптации гипермедиа и их применение при создании виртуального музея истории информатики в Сибири // Материалы XL Междунар. науч. студенческой конф. «Студент и научно-технический прогресс». Новосибирск, 2002. С. 173-174.
13. Волянская Т.А. Методы и технологии адаптивной гипермедиа // Современные проблемы конструирования программ. Новосибирск, 2002. С. 38-68.
14. Волянская Т.А., Касьянов В.Н., Несговорова Г.П. Адаптивная гипермедиа и ее использование при создании виртуального музея истории информатики в Сибири // Материалы Пятой международной конференции «Перспективы систем информатики» PSI'03. Новосибирск, 2003. С. 10-12.
15. Горшков С. Введение в онтологическое моделирование. ООО ТриниДата, 2018. 165 с.
16. Добров Б.В., Иванов В.В., Лукашевич Н.В., Соловьев В.Д. Онтологии и тезаурусы: модели, инструменты, приложения. М.: Бином. Лаборатория знаний, 2009. 173 с.
17. Иванов П., Иванова Р. Использование интеллект-карт в учебном процессе: Учебно-методическое пособие. LAP LAMBERT Academic Publishing, 2018. 68 с.
18. Касьянов В.Н. Музеи и Интернет: новая виртуальная реальность // Вычислительные технологии, 2008, Том 13, С. 239-247.
19. Касьянов В.Н. Музеи и Интернет: новые возможности // Информационные технологии в гуманитарных исследованиях. Новосибирск: Новосибирский госуниверситет, 2006. Выпуск 10. С. 88- 96.

20. Касьянов В.Н. Музеи и Интернет // Современные информационные технологии и письменное наследие: от древних текстов к электронным библиотекам. Казань: Изд-во Казанского гос. ун-та, 2008. С.132-135.
21. Касьянов В.Н, Касьянова Е.В. Адаптивные системы и методы дистанционного обучения // Информационные технологии в высшем образовании. 2004. Т.1, № 4. С. 40-60.
22. Касьянов В.Н., Несговорова Г.П., Волянская Т.А. Виртуальный музей истории информатики в Сибири // Проблемы программирования, 2003, N. 4, С. 82-91.
23. Касьянова Е.В. Адаптивные методы и средства поддержки дистанционного обучения программированию. Новосибирск: ИСИ СО РАН, 2007. 170 с.
24. Касьянова Е.В. Методы и средства обучения программированию в вузе // Образовательные ресурсы и технологии, 2016, № 2, С. 23-30.
25. Лапшин В.А. Онтологии в компьютерных системах. М.: Научный мир, 2010. 247 с.
26. Митрофанова О.А., Константинова Н.С. Онтологии как системы хранения знаний // Всероссийский конкурсный отбор обзорно-аналитических статей по приоритетному направлению «Информационно-телекоммуникационные системы», 2008. 54 с.
27. Муромцев Д.И. Концептуальное моделирование знаний в системе Concept Map. СПб: СПб ГУ ИТМО, 2009. 83 с.
28. Поллак Г.А. Современные технологии анализа информации: учебное пособие. Челябинск: Издательский центр ЮУрГУ, 2013. 115 с.
29. Anh N.V., Ha N.V., Dam H.S. Constructing a Bayesian Belief Network to Generate Learning Path in Adaptive Hypermedia System // Journal of Computer Science and Cybernetics. 2008. V. 24, N. 11. P. 12-19.
30. Ausubel, D. Educational Psychology: A Cognitive View. Holt, Rinehart & Winston, New York, 1968. 685 p.
31. Ausubel, D. The Acquisition and Retention of Knowledge: A Cognitive View. Kluwer Academic Publishers, Boston, 2000. 212 p.
32. Beaumont I., Brusilovsky P. Educational applications of adaptive hypermedia // Human-Computer Interaction, Proc. of Interact'95, Lillehammer, Norway. London, Chapman & Hall. 1995. P. 410-414.
33. Benyon D., Murray D. Adaptive systems: From intelligent tutoring to autonomous agents // Knowledge-Based Systems. 1993. V. 6, N. 4. P. 197-219.
34. Bhaskar M., Das M.M., Chithralekha T., Sivasathya S. Genetic Algorithm Based Adaptive Learning Scheme Generation For Context Aware E-Learning // IJCSE International Journal on Computer Science and Engineering. 2010. V. 2, N4. P. 1271-1279.
35. Bloom B.S. Taxonomy of educational objectives: The classification of educational goals: Handbook I, cognitive domain. New York, Toronto: Longmans, Green, 1956. 207 p.
36. Bontchev B., Vassileva D., Chavkova B., Mitev V. Architectural design of a software engine for adaptation control in the ADOPTA elearning platform // Proc. of the International Conference on

- Computer Systems and Technologies and Workshop for PhD Students in Computing, ACM, 2009. P. 1-6.
37. Brusilovsky P. Adaptive educational systems on the world-wide-web: A review of available technologies // Proc. of Workshop "WWW-Based Tutoring" at 4th International Conference on Intelligent Tutoring Systems (ITS'98). San Antonio, 1998.
  38. Brusilovsky P. Methods and Techniques of Adaptive Hypermedia // User Modeling and User Adapted Interaction. 1996. V. 6, N. 2-3. P. 87-129.
  39. Brusilovsky P., Nijhawan H. A framework for adaptive e-learning based on distributed re-usable learning activities // Proc. of World Conference on E-Learning, E-Learn 2002. Montreal, Canada, AACE, 2002. P. 154-161.
  40. Brusilovsky P., Schwarz E., Weber G. ELM-ART: An intelligent tutoring system on World Wide Web // Intelligent Tutoring Systems, Springer, 1996. P. 261-269.
  41. Brusilovsky P., Vassileva J. Course sequencing techniques for large-scale web-based education // International Journal of Continuing Engineering Education and Life Long Learning. 2003. V. 13, N. 1. P. 75-94.
  42. Buzan, Tony&Barry. The Mind Map Book. BBC:London, 2003. 277 p.
  43. Cañas, A.J., Reiska, P., Novak, J.D. Is My Concept Map Large Enough? // Innovating with Concept Mapping. CMC 2016. Communications in Computer and Information Science, vol. 635. Springer, Cham, 2016.
  44. Carchiolo V., Longheu A., Malgeri M., Mangioni G. Automatic generation of learning paths // Proc. of the 10th IEEE International Conference on Electronics, Circuits and Systems ICECS 2003, IEEE. 2003. V. 3. P. 1236-1239.
  45. Carr B., Goldstein I.P. Overlays. A theory of modeling for computer-aided instruction, AI Lab Memo 406 / Massachusetts Institute of Technology, Cambridge. Massachusetts, 1977.
  46. Chang M., Chang A., Heh J.S., Liu T.C. Contextaware learning path planner // WSEAS Transactions on Computers. 2008. V. 7, N. 4. P. 316-325.
  47. Chapelle C., Mizuno S. Student's strategies with learner-controlled CALL // Calico Journal. 1989. V. 7, N. 2, P. 25-47.
  48. Conlan O., Wade V., Bruen C., Gargan M. Multi-model, Metadata Driven Approach to Adaptive Hypermedia Services for Personalized eLearning // Adaptive Hypermedia and Adaptive Web-Based Systems / Eds. Bra P. D., Brusilovsky P., and Conejo R. (Lecture Notes in Computer Science). Springer, 2002. V. 2347. P. 100-111.
  49. Dagger D., Wade V., Conlan O. Developing Adaptive Pedagogy with the Adaptive Course Construction Toolkit (ACCT) // Proc. of the Third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, AH2004 / Eds. De Bra P. and Nejd W. Berlin: Springer Verlag, Eindhoven, The Netherlands, 2004. P. 55-64.

50. Dagger D., Wade V., Conlan O. Personalisation for all: Making adaptive course composition easy // *Educational Technology & Society*. 2005. V. 8, N. 3. P. 9-25.
51. Dijkstra S., Krammer H.P., Van Merriënboer J.G. *Instructional models in computer-based learning environments*. – Springer-Verlag New York, Inc. Secaucus, NJ, USA, 1992. 516 p.
52. Dolenc K., Aberšek B. TECH8 intelligent and adaptive elearning system: Integration into Technology and Science classrooms in lower secondary schools // *Computers & Education*. 2015. V. 82. P. 354-365.
53. Essalmi F., Ayed L.J.B., Jemni M., Kinshuk, Graf S. A fully personalization strategy of E-learning scenarios // *Computers in Human Behavior*. 2010. V. 26, N. 4. P. 581-591.
54. Felder R.M., Silverman L.K. Learning and teaching styles in engineering education // *Engineering education*. 1988. V. 78, N. 7. P. 674-681.
55. Gascuena J.M., Fernandez-Caballero A., Gonzalez P. Domain ontology for personalized e-learning in educational systems // *Proc. of the Sixth IEEE International Conference on Advanced Learning Technologies*. 2006. P. 456-458.
56. Gavignet E. *Environnement de conception de système d'apprentissage: une modélisation de la connaissance pédagogique: PhD thesis, de l'Université de Nancy 1*. 1991.
57. Grandbastien M. Teaching expertise is at the core of ITS research // *International journal of artificial intelligence in education*. 1999. V. 10, N. 3-4. P. 335-349.
58. Grandbastien M., Gavignet E. ECSA: An Environment to Design and Instantiate Learning Material // *Handbook of Design and Production of Multimedia and Simulation-based Learning Material / Eds. Jong T. and Sarti L. Kluwer Academic, Publishers, Dordrecht, The Netherlands, 1994. P. 31-44.*
59. Gruber Thomas R. A translation approach to portable ontology specifications // *Knowledge Acquisition*. 1993. V. 5. P. 199-220.
60. Gruber Thomas R. Toward principles for the design of ontologies used for knowledge sharing // *International Journal of Human-Computer Studies*. 1995. Vol.43, № 5-6. P. 907–928.
61. Gruber T. *Ontology // Encyclopedia of Database Systems, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag, 2009.*
62. Guarino N. Formal Ontology and Information Systems // *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, Trento, Italy. IOS Press, Amsterdam, 1998. P. 3-15.*
63. Guriano N. Understanding, Building, and Using Ontologies // *International Journal of Human and Computer Studies*. 1997. Vol. 46. P. 293-310.
64. Grubišić A., Stankov S., Žitko B. Adaptive Courseware: A Literature Review // *Journal of Universal Computer Science*. 2015. V. 21, N. 9. P. 1168-1209.
65. Henze N., Nejdil W. Adaptivity in the KBS Hyperbook System // *Proc. of the 2nd Workshop on User Modeling and Adaptive Systems on the WWW. Toronto, Canada, 1999. P. 67-74.*
66. Honey P., Mumford A. *The manual of learning styles. Maidenhead, Peter Honey, 1992.*

67. Karampiperis P., Sampson, D. Adaptive instructional planning using ontologies // Proc. of the IEEE International Conference on Advanced Learning Technologies, 2004. P. 126–130.
68. Karampiperis P., Sampson D. Adaptive learning resources sequencing in educational hypermedia systems // Educational Technology & Society. 2005. V. 8, N. 4. P. 128-147.
69. Kasyanov V.N. An open adaptive virtual museum of informatics history in Siberia // Proc. of the 20th IFIP World Computer Congress // IFIP International Federation for Information Processing. Boston: Springer, 2008. V. 266. History of Computing and Education 3 (HCE 3). P. 129-146.
70. Kasyanov V.N. An open adaptive virtual museum of information history in Siberia // IFIP Advances in Information and Communication Technology. Berlin: Springer, 2011. Vol. 357. P. 194-200.
71. Kasyanov V. SVM – Siberian Virtual Museum of Informatics History // Innovation and the Knowledge Economy: Issues, Applications, Case Studies. Amsterdam: IOS Press, 2005. Part 2. P.1014-1021.
72. Kasyanov V.N., Kasyanova E.V. A Web-based system for distance learning of programming // Proc. of the European Computing Conference / Lecture Notes in Electrical Engineering. Springer, 2009. V. 27. P. 453-462.
73. Kasyanov V.N., Kasyanova E.V. Graph- and cloud-based tools for computer science education // Lecture Notes of Computer Science, 2015, Vol. 9395, P. 41-54.
74. Kasyanov V.N., Kasyanova E.V. Methods and Tools for Teaching Programming at University // Journal of Physics: Conference Series, 2020, Vol. 1603, 012021.
75. Kasyanov V., Kasyanova E. WAPE — a system for distance learning of programming // Learning to Live in the Knowledge Society: IFIP 20th World Computer Congress. Boston: Springer, 2008. P. 355-356. (IFIP International Federation for Information Processing, V. 261).
76. Latham A., Crockett K., McLean D. An adaptation algorithm for an intelligent natural language tutoring system // Computers & Education. 2014. Vol 71. P.97-110.
77. Lin T. Cognitive Trait Model for Adaptive Learning Environments: PhD Thesis, Massey University. Palmerston North, New Zealand, 2007.
78. Magoulas G.D., Papanikolaou Y., Grigoriadou M. Adaptive web based learning: accommodating individual differences through system's adaptation // British Journal of Educational Technology. 2003. V. 34, N. 4. P. 511-527.
79. Márquez J.M., Ortega J.A., Gonzalez-Abril L., Velasco F. Creating adaptive learning paths using Ant Colony Optimization and Bayesian Networks // Proc. of the IEEE World Congress on Computational Intelligence IJCNN 2008. 2008. P. 3834-3839.
80. Melis E., Gogvadze G., Libbrecht P., Ullrich C. Activemath – a learning platform with semantic web features // Semantic Web Technologies for e-Learning / Eds. Dicheva D., Mizoguchi R. and Greer J. (Eds.). IOS Press, The Future of Learning. 2009. P. 159-177.
81. Mödritscher F. Adaptive E-Learning Environments: Theory, Practice, and Experience. Verlag Dr. Müller, 2008.

82. Mohan P., Greer J., McCalla G. Instructional planning with learning objects // Knowledge Representation and Automated Reasoning for E-Learning Systems. 2003. P. 52-58.
83. Muñoz- Merino P.J., Fernández Molina M., Muñoz-Organero M., Delgado Kloos C. An adaptive and innovative question-driven competition-based intelligent tutoring system for learning // Expert Systems with Applications. 2012. V. 39, N. 8. P. 6932-6948.
84. Novak, J.D. & Gowin, D.B. Learning How To Learn. Cambridge University Press: New York, 1984. 216 p.
85. Novak J.D., Cañas A.J. The Theory Underlying Concept Maps and How To Construct and Use Them. Florida Institute for Human and Machine Cognition, 2008. (Tech. Rep. / IHMC CmapTools 2006-01 Rev 01-2008).
86. Novak, J.D., Cañas, A.J. Theoretical Origins of Concept Maps, How to Construct Them, und Uses in Education. Reflecting Education Vol3 No1, 2007.
87. O Keeffe I., Brady A., Conlan O., Wade V. Just-in-time generation of pedagogically sound, context sensitive personalized learning experiences // International Journal on e-Learning. 2006. V. 5, N1. P. 113-127.
88. Oppermann R., Rasher R. Adaptability and adaptivity in learning systems // Knowledge Transfer. 1997. Vol 2. P. 173-179.
89. Özyurt Ö., Özyurt H., Baki A., Güven B., Karal H. Evaluation of an adaptive and intelligent educational hypermedia for enhanced individual learning of mathematics: A qualitative study // Expert Systems with Applications. 2012. V. 39, N. 15. P. 12092-12104.
90. Papanikolaou K.A., Grigoriadou M., Kornilakis H., Magoulas G.D. Personalizing the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE // User Modeling and User-Adapted Interaction. 2003. V. 13, N. 3. P. 213-267.
91. Park O., Lee J. Adaptive instructional systems // Handbook of Research on Educational Communications and Technology, 3rd ed / Eds. Spector J.M., Driscoll M.P., Van Merriënboer J., and Merrill M.D. 2008. P. 469-484.
92. Pukkhem N., Evens M.W., Vatanawood W. The Concept Path Combination Model for Supporting a Personalized Learning Path in Adaptive Educational Systems // Proc. of the 2006 International Conference on e-Learning, e-Business, Enterprise Information Systems, e-Government, and Outsourcing (EEE'06). 2006. P. 10-16.
93. Shute V.J., Zapata-Rivera D. Adaptive technologies // Handbook of research on educational communications and technology. 2008. P. 277-294.
94. Smith M.K., Welty C., McGuinness D. OWL Web Ontology Language Guide. 10.02.2004. [Электронный ресурс]. URL: <http://www.w3.org/TR/owl-guide/>
95. Specht M., Kravcik M., Pesin L., Klemke R. Authoring adaptive educational hypermedia in WINDS // Proc. of ABIS2001. Dortmund, Germany, 2001. P. 1-8.

96. Specht M., Oppermann R. ACE-adaptive courseware environment // *The New Review of Hypermedia and Multimedia*. 1998. V. 4, N. 1. P. 141-161.
97. Specht M., Weber G., Heitmeyer S., Schöch V. AST: Adaptive WWW-courseware for statistics // *Proc. of Workshop 'Adaptive Systems and User Modeling on the World Wide Web' at 6th International Conference on User Modeling, UM97*. Chia Laguna, Sardinia, Italy, 1997. P. 91-95.
98. Ullrich C. Pedagogically founded courseware generation for web-based learning: an HTN-planning-based approach implemented in PAIGOS // *Lecture Notes in Computer Science*. 2008. V. 5260. P. 111-167.
99. Uschold M., Gruninger M. *Ontologies: Principles, Methods and Applications* // *Knowledge Engineering Review*. 1996. Vol.11, № 2. P. 93-155.
100. Van Marcke K. A generic tutoring environment // *The European Conference on Artificial Intelligence*. 1990. P. 655-660.
101. Van Marcke K. *Instructional expertise* // *Proc. of the Second International Conference on Intelligent Tutoring Systems*. Springer, 1992. P. 234-243.
102. Van Marcke K. GTE: An epistemological approach to instructional modelling // *Instructional Science*. 1998. V. 26, N. 3. P. 147-191.
103. Vassileva J. Dynamic CAL-courseware generation within an ITS-shell architecture // *Proc. of the International Conference on Computer Assisted Learning ICCAL / Lecture Notes in Computer Science*. 1992. V. 602. P. 581-591.
104. Vassileva J. Dynamic courseware generation: at the cross point of CAL, ITS and authoring // *Proc. of ICCE*. 1995. P. 290-297.
105. Vassileva J. DCG+ GTE: Dynamic courseware generation with teaching expertise // *Instructional Science*. 1998. V. 26, N. 3. P. 317-332.
106. Vassileva J., Wasson B. *Instructional planning approaches: From tutoring towards free learning* // *Proc. of EuroAIED96*. 1996. P. 1-8.
107. Weber G. *Episodic Learner Modeling* // *Cognitive Science*. 1996. Vol.20, N. 2. P. 195-236.
108. Weber G., Kuhl H.C., Weibelzahl S. Developing adaptive internet based courses with the authoring system NetCoach // *Hypermedia: Openness, Structural Awareness, and Adaptivity*. 2002. P. 222-223.
109. Weber G., Moellenberg A. *ELM-Programming- Environment: A Tutoring System for LISP Beginners* // *Cognition and Computer Programming* / Eds. Wender K. F., Schmalhofer F., and Böcker H.-D. Norwood, NJ: Ablex, 1995. P. 373-408.
110. Wiley D.A. Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy // *The instructional use of learning objects*. 2000. V. 2830, N. 435. P. 1-35.
111. Zhu F., Cao J. Learning activity sequencing in personalized education system // *Wuhan University Journal of Natural Sciences*. 2008. V. 13, N. 4. P. 461-465.



УДК 004.89:616-073.7

## **Разработка системы мониторинга электрокардиограмм на основе микроконтроллера ESP8266**

*Айсайти Байшань (Новосибирский государственный университет)*

*Загорулько Ю.А. (Институт систем информатики СО РАН, Новосибирский  
государственный университет)*

Статья посвящена разработке прототипа системы мониторинга электрокардиограмм (ЭКГ) на основе микроконтроллера ESP8266. Система использует датчик AD8232 для сбора данных ЭКГ, которые передаются на сервер по протоколу Wi-Fi. На серверной стороне данные обрабатываются с использованием фреймворка PyTorch и модели LSTM для анализа в реальном времени. Основное внимание уделено проектированию прототипа системы, деталям сбора и предварительной обработки данных, а также применению модели LSTM. Результаты тестирования системы показывают, что она способна эффективно отслеживать сигналы ЭКГ и выявлять аномалии с высокой точностью и в реальном времени.

**Ключевые слова:** мониторинг, электрокардиограмма, аритмия, LSTM, глубокое обучение, машинное обучение.

### **1. Введение**

Электрокардиограмма (ЭКГ) является ключевым инструментом для оценки состояния сердечно-сосудистой системы человека. Традиционные устройства ЭКГ часто громоздки, дороги и неудобны для повседневного использования. С развитием технологий Интернета вещей (IoT) и глубокого обучения появилась возможность создания портативных и доступных систем мониторинга ЭКГ. В данной работе предлагается система на основе микроконтроллера ESP8266, использующая датчик AD8232 и модель LSTM для сбора, передачи и анализа данных ЭКГ в реальном времени. Целью исследования является разработка прототипа системы, сочетающей низкую стоимость, портативность и высокую точность анализа.

В последние годы исследователи разработали множество портативных систем мониторинга ЭКГ, включая решения на базе Arduino-Uno [4] и мобильных приложений. Глубокое обучение также активно применяется для анализа сигналов ЭКГ, особенно

рекуррентные нейронные сети (Recurrent neural network, RNN) [11] и их разновидность — долгая краткосрочная память (Long short-term memory; LSTM) [7], которые демонстрируют высокую эффективность в обработке временных рядов. Настоящее исследование объединяет эти подходы, предлагая новую систему мониторинга ЭКГ с использованием современных технологий.

В аналогичной системе мониторинга ЭКГ [10] использовался микроконтроллер Arduino Uno. В сравнении с ним, ESP8266 обладает рядом преимуществ, включая встроенный модуль Wi-Fi, более высокую тактовую частоту, увеличенный объем памяти и поддержку низкого энергопотребления, что делает его более подходящим для разработки беспроводных и энергоэффективных систем мониторинга ЭКГ (Таблица 1).

Таблица 1. Сравнение характеристик ESP8266 и Arduino Uno

	<b>ESP8266</b>	Arduino Uno
Процессор	32-битный, 80/160 МГц	8-битный, 16 МГц
Оперативная память	128 КБ	2 КБ
Память для кода (Flash)	4 МБ	32 КБ
Wi-Fi	Встроенный модуль	Отсутствует (требуется внешний модуль)
Рабочее напряжение	3.3 v	5 v
Потребление энергии	15 мкА – 400 мА	45 мА – 80 мА
Стоимость	Низкая	Высокая
Размеры	Компактные	Крупные

## 2. Введение в ЭКГ

Электрокардиография — это неинвазивный метод исследования, используемый для регистрации электрической активности сердца. При проведении электрокардиографии электроды прикрепляются к определённым участкам тела (таким как грудная клетка, руки и ноги), что позволяет зафиксировать слабые электрические сигналы, возникающие при каждом сердечном сокращении, и отобразить их в виде волн на ЭКГ [1].

Каждое сердечное сокращение инициируется электрическим импульсом, который начинается в синусно-предсердном узле и последовательно распространяется через предсердия, атриовентрикулярный узел, пучок Гиса, его правую и левую ножки, а затем по волокнам Пуркинье, вызывая сокращение миокарда. Электрокардиограмма (ЭКГ)

регистрирует изменения этих электрических сигналов (Рис. 1), отражая электрическую активность сердца.

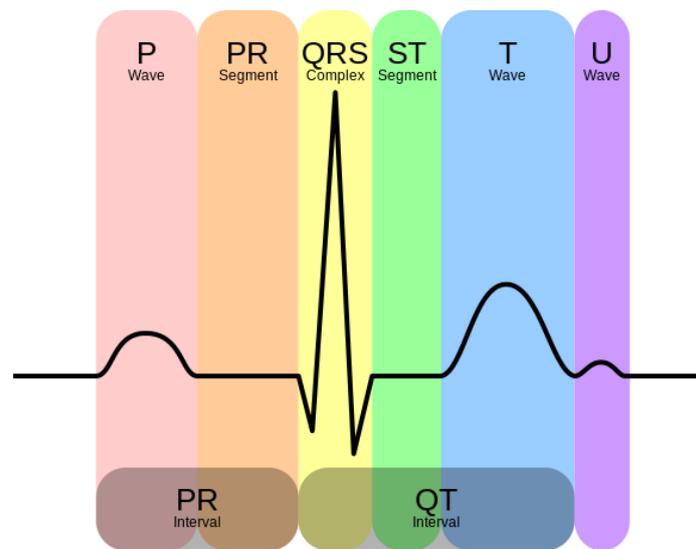


Рис. 1. Форма волны электрокардиограммы<sup>1</sup>

Стандартная ЭКГ обычно включает следующие виды волн:

- Волна P: представляет деполяризацию предсердий, то есть электрическую активность, связанную с их сокращением.
- Комплекс QRS: отражает деполяризацию желудочков, связанную с их сокращением.
- Волна T: представляет реполяризацию желудочков, то есть восстановление их электрического состояния.

Форма, продолжительность и амплитуда этих волн предоставляют важную информацию о ритме и проводимости сердца, а также о состоянии миокарда.

С помощью ЭКГ можно выявить широкий спектр кардиологических заболеваний, в частности, аритмию, брадикардию, тахикардию, стенокардию, гипертрофию сердечных отделов, аневризму сердца, экстрасистолию, тромбоз легочной артерии, миокардит, инфаркт миокарда и др.

Так, одним из главных признаков инфаркта миокарда на ЭКГ является смещение ST-сегмента. ЭКГ здорового сердца имеет стабильный уровень изолинии ST-сегмента. В случае инфаркта, наблюдается подъем или депрессия ST-сегмента. Изменения этого элемента часто свидетельствуют о ранних стадиях инфаркта и позволяют определить его местоположение.

<sup>1</sup> <https://learn.sparkfun.com/tutorials/ad8232-heart-rate-monitor-hookup-guide/all>

### 3. Архитектура системы

Система мониторинга (Рис. 2) включает подсистему сбора данных, подсистему хранения и анализа данных, а также подсистему, отвечающую за визуализацию данных и результатов анализа ЭКГ и информирование пациента о состоянии его здоровья.

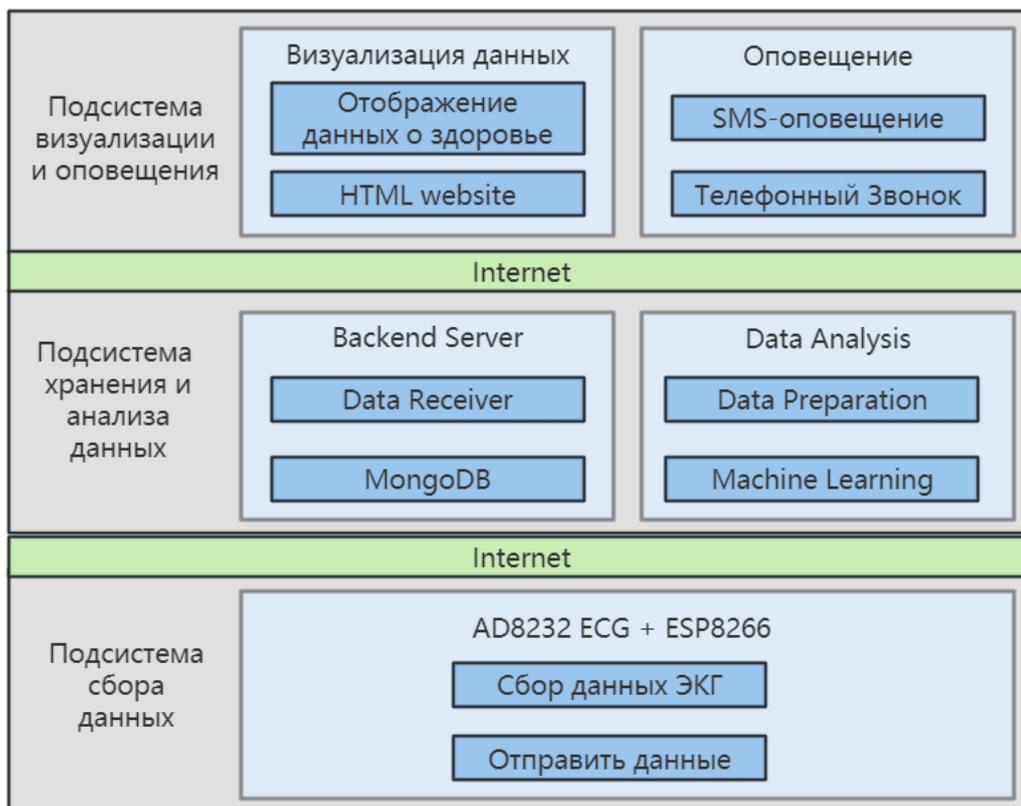


Рис. 2. Архитектура системы

**Функционирование системы поддерживается следующими компонентами:**

- Плата ESP8266: отвечает за считывание данных ЭКГ с датчика AD8232 и передачу их на сервер по Wi-Fi (см. Рис.3).
- Датчик AD8232: используется для сбора сигналов ЭКГ (см. Рис.4).
- Сервер: работает с приложением Nodejs [8] для приема данных, их предварительной обработки и анализа с помощью предобученной модели LSTM. Обработанные данные сохраняются в базе данных MongoDB [2]. Визуализация осциллограмм ЭКГ выполняется в реальном времени с помощью фронтэнда Vue [5].

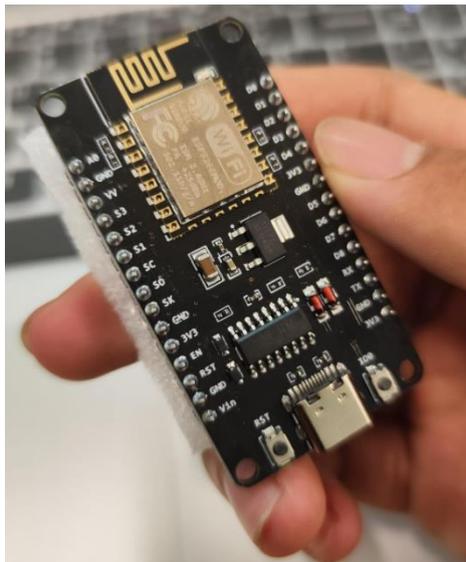


Рис. 3. Общий вид платы ESP8266.

ESP8266 — это недорогой микроконтроллер с поддержкой Wi-Fi, широко используемый в проектах Интернета вещей (IoT) [9].

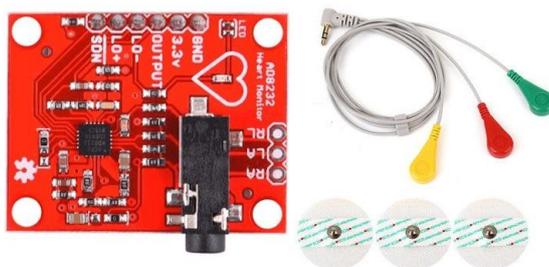


Рис. 4. Модуль подготовки сигналов ЭКГ AD8232.<sup>2</sup>

AD8232 — это модуль подготовки сигналов ЭКГ, специально разработанный для портативных устройств мониторинга ЭКГ с низким энергопотреблением, который объединяет в себе усилитель, фильтр и функцию обнаружения отсоединения электрода [6].

## 4. Сбор данных

ESP8266 считывает данные ЭКГ с датчика AD8232 через аналого-цифровой преобразователь (АЦП). Частота дискретизации составляет 250 Гц, что означает сбор одного значения каждые 4 миллисекунды (см. Рис.5).

<sup>2</sup> <https://electronicsworkshops.com/2022/12/06/iot-based-ecg-and-heart-rate-monitoring-with-ad8232-ecg-sensor-esp8266/>

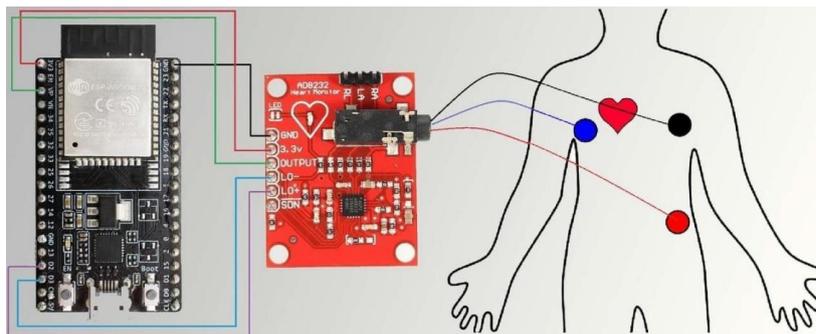


Рис. 5. Схема подключения модуля AD8232 к плате ESP8266

Собранные данные отправляются на сервер в формате JSON с использованием HTTP-запросов POST. Такой подход обеспечивает стабильную и надежную передачу данных даже при ограниченной пропускной способности сети.

## 5. Предварительная обработка данных

После получения данных измерений ЭКГ сервер выполняет их предварительную обработку. На Рис. 6 показаны наиболее важные точки ЭКГ.

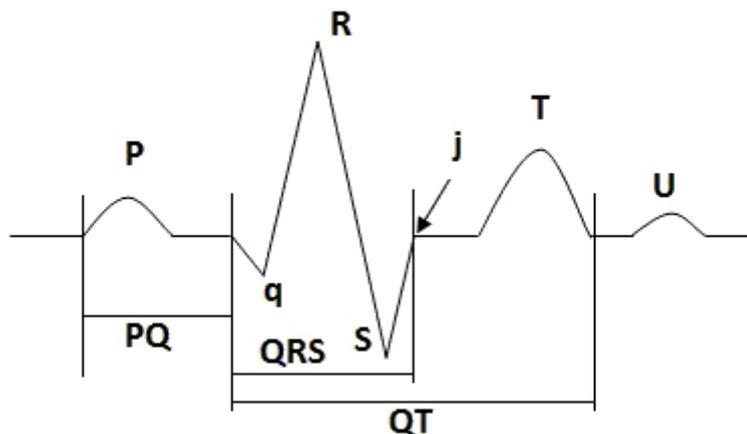


Рис. 6. Характерные точки ЭКГ

Предварительная обработка данных включает следующие шаги:

- **Фильтрация:** применяется полосовой фильтр Баттерворта с диапазоном частот 0.5–40 Гц для удаления шума, включая артефакты движения и высокочастотные помехи.
- **Нормализация:** данные масштабируются в диапазоне  $[0, 1]$  для соответствия требованиям модели машинного обучения. Обработанные данные формируются в последовательности фиксированной длины (например, 1000 временных шагов), что соответствует входным данным модели LSTM.
- **Вычисление сердечного ритма:** для полученного сигнала электрокардиограммы,

имеющего в момент времени  $t$  значение  $V(t)$ , обнаружение пиков  $R$  обычно осуществляется с использованием порогового значения:

- Определение порога: устанавливается пороговое значение  $T$ , которое обычно определяется на основе опыта или статистического анализа предварительно обработанных данных.
- Обнаружение пиков: если  $V(t) > T$  и в определённом интервале времени до и после момента  $t$  нет значений сигнала, превышающих  $V(t)$ , то считается, что в момент  $t$  зафиксирован пик  $R$ . После обнаружения последовательности пиков  $R$  во временные моменты  $t_1, t_2, \dots, t_n$ , можно вычислить интервалы  $RR$ :

$$RR_i = t_i - t_{i-1}$$

- Затем рассчитывается частота сердечных сокращений (ЧСС) по формуле:

$$\text{ЧСС}_i = \frac{60}{RR_i}$$

Здесь  $RR_i$  выражается в секундах, а  $\text{ЧСС}_i$  — в ударах в минуту (уд/мин).

- **Сохранение данных:** данные сохраняются в базе данных MongoDB в соответствии с заданным форматом (см. Рис.7 и Рис.8).

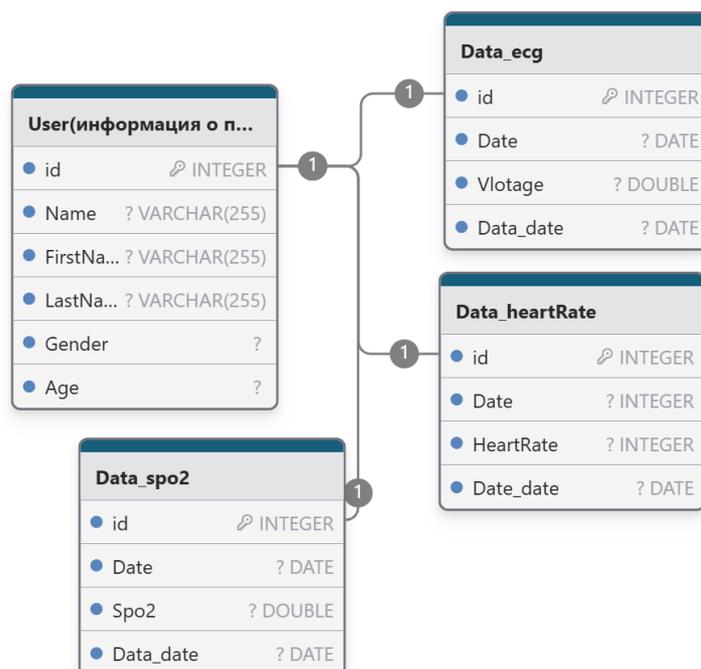


Рис. 7. Схема данных

```
_id: ObjectId('681a35ac07ee1ff1b09b8438')
timestamp: 1970-01-01T00:00:23.761+00:00
▶ data: Array (250)
  heartRate: 144
  __v: 0
```

Рис. 8. Пример данных

## 6. Модель LSTM

Для анализа сигналов ЭКГ используется модель LSTM, реализованная в фреймворке машинного обучения PyTorch. Архитектура модели включает:

- **Входной слой:** принимает тензор размером (1, 1000, 1), где 1000 — длина временной последовательности.
- **Слой LSTM:** два слоя LSTM с размером скрытого состояния 128.
- **Полносвязный слой:** преобразует выход LSTM в одно значение, представляющее результат классификации (например, нормальный или аномальный сигнал).

Модель была обучена на наборе данных PTB-XL с использованием функции потерь бинарной кросс-энтропии и оптимизатора Adam [3]. Процесс обучения проводился на сервере с использованием графического процессора (GPU) для ускорения вычислений.

## 7. Эксперименты и результаты

Эксперименты проводились в симулированной среде с использованием набора данных PTB-XL [12], содержащего разнообразные записи ЭКГ (Рис.9). Модель обучалась и тестировалась на сервере с GPU, что позволило сократить время обработки данных.

```
# Step 1: Load and preprocess the PTB-XL dataset
data_path = './ptb-xl/' # Update with your dataset path
metadata = pd.read_csv(os.path.join(data_path, 'ptbx1_database.csv'))
```

Рис. 9. Загрузка набора данных PTB-XL

Набор данных PTB-XL — это мощная и всеобъемлющая база данных электрокардиограмм, содержащая 21 837 записей 12-канальных ЭКГ продолжительностью 10 секунд, полученных от 18 885 пациентов, с 71 экспертными комментариями и обширными метаданными.

```
class RNNModel(nn.Module):
    def __init__(self, input_size, hidden_size, num_layers, output_size):
        super(RNNModel, self).__init__()
        self.lstm = nn.LSTM(input_size, hidden_size, num_layers, batch_first=True)
        self.fc = nn.Linear(hidden_size, output_size)

    def forward(self, x):
        out, _ = self.lstm(x)
        out = self.fc(out[:, -1, :]) # Use the last time step
        return out
```

Рис. 10. Определение модели RNN

Модуль `torch.nn.RNN` в PyTorch представляет собой готовый класс для создания RNN-слоев модели (Рис.10). RNN — это архитектура нейронной сети, специально предназначенная для обработки временных рядов или последовательных данных, которая позволяет улавливать временные зависимости в данных.

```
# Step 5: Save the model
torch.save(model.state_dict(), 'rnn_model0504_1.pth')
```

Рис. 11. Завершение машинного обучения и получение весов модели

```
Epoch [8/10], Loss: 0.4265
step3 fine
Epoch [9/10], Loss: 0.3910
step3 fine
Epoch [10/10], Loss: 0.3792
Test Accuracy: 83.81%
```

Рис. 12. Оценка модели

Для обучения модели использовалось 80% данных из набора PTB-XL (Рис. 11), а оставшиеся 20% были выделены для её оценки. Точность модели составила 83.81% (см. Рис.12).

## 8. Визуализация результатов мониторинга

Визуализация осциллограмм ЭКГ выполняется в реальном времени с помощью фронтенда Vue (Рис. 13).

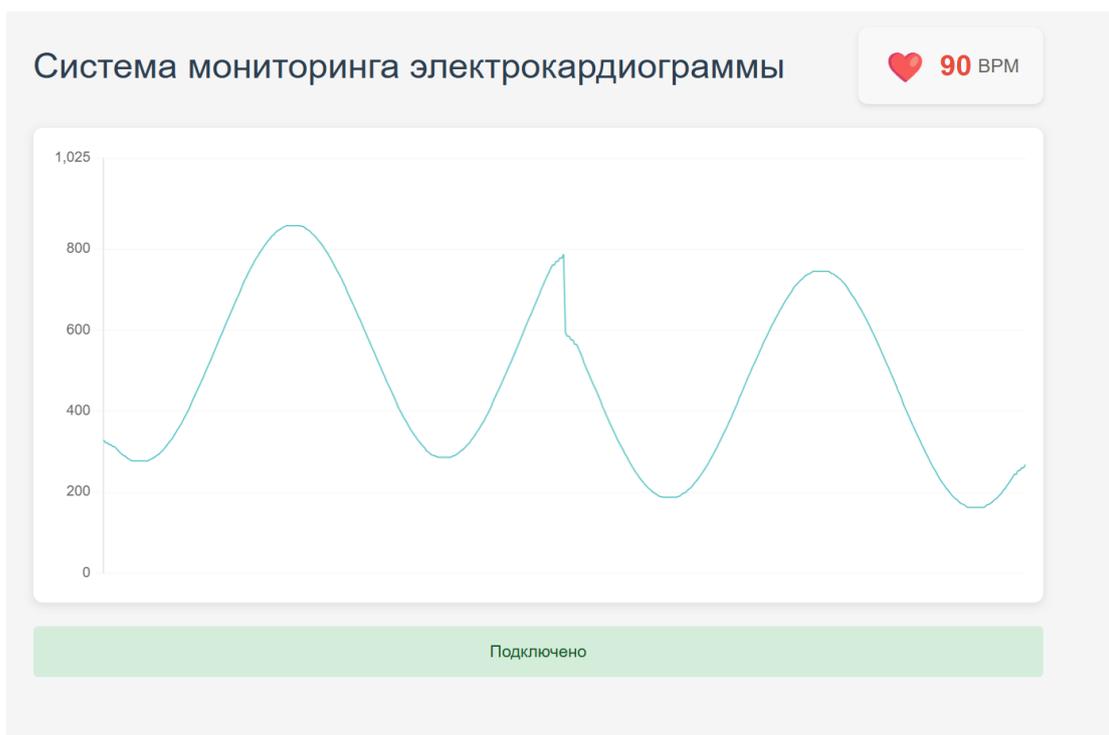


Рис. 13. Волновые формы ЭКГ

При обнаружении аномального сердечного ритма (брадикардия или тахикардия) система предоставляет пользователю рекомендации по здоровью и автоматически отправляет уведомление врачу через SMS (см. Рис. 14 и 15).

The screenshot shows the 'Система мониторинга электрокардиограммы' (ECG Monitoring System) interface with a heart rate of 40 BPM. A warning message is displayed: 'обратите внимание!' (Attention!). The message text reads: 'Частота сердечных сокращений в состоянии покоя составляет менее 60 ударов в минуту. Такие факторы, как аномалии в проводящей системе сердца, гипотиреоз и некоторые лекарственные препараты (например, бета-блокаторы), могут способствовать развитию брадикардии. Пожалуйста, поддерживайте связь с вашим врачом, носите с собой мобильный телефон и следите за уровнем заряда батареи вашего мобильного телефона'. A red button labeled 'свертывание' (Collapse) is located to the right of the message. At the bottom, a green bar contains the text 'Подключено' (Connected).

Рис. 14. Низкая частота сердечных сокращений

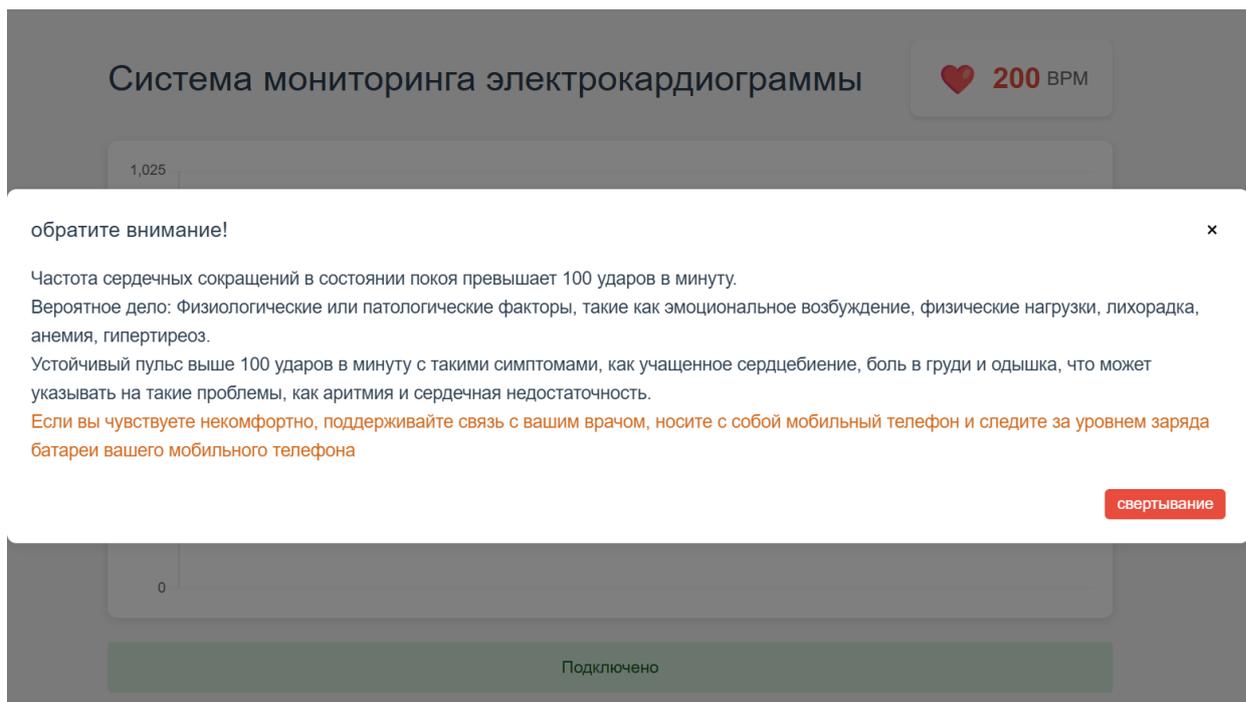


Рис. 15. Чрезмерная частота сердечных сокращений

## Заключение

Проведён анализ отечественных и зарубежных исследований в области систем мониторинга ЭКГ, что позволило определить современное состояние технологий и обосновать выбор архитектуры беспроводной носимой системы ЭКГ-мониторинга.

Разработан прототип системы мониторинга ЭКГ на основе микроконтроллера ESP8266 с использованием датчика AD8232 и модели LSTM.

Разработана аппаратная часть системы, включающая модуль сбора ЭКГ-сигналов на основе датчика AD8232. Сигналы обрабатываются встроенными усилителями и фильтрами (высокочастотными и низкочастотными), после чего микроконтроллер ESP8266 выполняет аналого-цифровое преобразование и цифровую фильтрацию. Обработанные данные сохраняются в базе данных MongoDB, а визуализация в реальном времени реализована с использованием фреймворка Vue.js.

Для обучения модели был использован набор данных PTB-XL. С применением фреймворка PyTorch и рекуррентной нейронной сети (RNN) проведено обучение модели, в результате чего достигнута точность классификации аномалий в 83,81 %.

В дальнейшем мы планируем улучшить программный код, оптимизировать архитектуру модели, более точно настроить гиперпараметры и увеличить объём обучающих данных.

**Благодарности.** Эксперимент проводился на рабочей станции института систем информатики им. А.П. Ершова СО РАН, оснащенной видеокартой с 16 ГБ памяти.

## Список литературы

1. Chandra S, Sharma A, Singh G K. Feature extraction of ECG signal // *Journal of Medical Engineering & Technology*, 2018. Vol. 42(4). P. 306–316.
2. Győrödi C, Győrödi R, Pecherle G, et al. A comparative study: MongoDB vs. MySQL // *2015 13th international conference on engineering of modern electric systems (EMES)*. IEEE, 2015. P. 1–6.
3. Hansen B. Heart Disease Classification using Transformers in PyTorch. 2021. URL: <https://towardsdatascience.com/heart-disease-classification-using-transformers-in-pytorch-8dbd277e079/>
4. Iskandar W J, Roihan I, Koestoer R A. Prototype low-cost portable electrocardiogram (ECG) based on Arduino-Uno with Bluetooth feature // *AIP Conference Proceedings*. AIP Publishing. 2019. P. 2193(1).
5. Kumpulainen T. Web application development with Vue.js. 2021.
6. Mendes Junior J J A, Campos D P, Biassio L C A V D, et al. AD8232 to biopotentials sensors: Open source project and benchmark // *Electronics*. 2023, Vol. 12(4): 833.
7. Saadatnejad S, Oveisi M, Hashemi M. LSTM-based ECG classification for continuous monitoring on personal wearable devices. *IEEE journal of biomedical and health informatics*, 2019. Vol. 24(2): P. 515–523.
8. Satheesh M, D'mello B J, Krol J. Web development with MongoDB and NodeJs // *Packt Publishing Ltd*, 2015.
9. Schwartz M. Internet of things with ESP8266 // *Packt Publishing Ltd*, 2016.
10. Setiawan M H, Sari N A, Prasetya W L, et al. Implementation of Heart Rate System using AD8232 and Arduino Microcontrollers // *Signal and Image Processing Letters*. 2020. Vol. 2(1). P. 36–44.
11. Singh S, Pandey S K, Pawar U, et al. Classification of ECG arrhythmia using recurrent neural networks // *Procedia computer science*. 2018. Vol. 132. P. 1290–1297.
12. Wagner, P., Strodthoff, N., Boussejot, RD. et al. PTB-XL, a large publicly available electrocardiography dataset // *Sci. Data*. 2020. Vol. 7:154. <https://doi.org/10.1038/s41597-020-0495-6>

УДК 004.912 + 004.8

## Связывание терминов из научных текстов с сущностями баз знаний

*Кузовлев Д.Д. (Институт систем информатики им. А.П. Ершова СО РАН),  
Батура Т.В. (Институт систем информатики им. А.П. Ершова СО РАН),  
Старцев П.А. (Институт систем информатики им. А.П. Ершова СО РАН)*

В данной статье предлагаются новые алгоритмы связывания научных терминов с сущностями Википедии и MeSH (Medical Subject Headings), работающие в условиях ограниченного количества данных. Алгоритм связывания с Википедией для части коллекции текстов на русском языке использует поисковую систему Википедии для генерации кандидатов и библиотеку spaCy для получения векторного представления текста. Подсчет семантической близости между описанием сущности из Википедии и термином из текста выполняется не только на основе самого научного термина, но и контекста, в котором он расположен. Для части коллекции по медицинской тематике, содержащей переводы с русского на английский язык, описан и реализован алгоритм связывания терминов, который опирается на словарь медицинских предметных рубрик MeSH. Результаты экспериментов показывают значения F1-меры 50.77 % для Википедии и 40.05 % для MeSH, что является хорошим показателем в условиях малого объема размеченных данных. Проведенное исследование подчеркивает необходимость развития специализированных русскоязычных баз знаний по аналогии с MeSH. Перспективным направлением является применение мультязычных моделей для кросс-лингвистического связывания, что особенно актуально для редких терминов. Полученные результаты могут применяться при создании интеллектуальных систем анализа научных текстов и автоматизированных научных ассистентов, что особенно важно для узких предметных областей.

**Ключевые слова:** обработка текстов, извлечение информации, связывание сущностей, поиск и ранжирование, Википедия, MeSH

### 1. Введение

Научные тексты всегда были и остаются основным источником новых знаний. Однако с увеличением объема публикуемых материалов возникает проблема эффективного извлечения и структурирования информации. Одной из ключевых задач в области обработки естественного языка является связывание сущностей (Entity Linking, EL), которая заключается в сопоставлении упоминаемых понятий в текстах, с соответствующими записями в базах знаний [1]. Данная статья посвящена исследованию

этой важной задачи для научных текстов, где в качестве терминов выступают слова или фразы, используемые в определенной предметной области для точного обозначения конкретных понятий, явлений или объектов; под сущностями подразумеваются элементы (записи) базы знаний. Решение этой задачи открывает новые возможности для создания интеллектуальных систем, способных автоматически анализировать научные публикации, извлекать ключевые концепции и связывать их с уже существующими структурированными знаниями.

Связывание терминов с сущностями баз знаний представляет собой сложную задачу, основные трудности которой связаны с многозначностью терминов, различиями в терминологии между дисциплинами, а также с необходимостью учета контекста использования терминов в тексте. Так, современные методы EL активно используют контекстуальное усиление (contextual augmentation) для улучшения качества связывания. Авторы [2] предлагают подход, основанный на больших языковых моделях (Large Language Model, LLM), которые генерируют дополнительные контексты для упоминаний сущностей, помогая разрешать неоднозначности. Метод включает динамическое расширение контекста запроса и семантическое ранжирование кандидатов, что особенно полезно для коротких или малоинформативных упоминаний. Важным преимуществом является адаптивность под разные области знаний, что делает подход универсальным для специализированных баз знаний.

В статье [3] представлен масштабируемый метод EL, работающий для 100 языков, включая редкие и малоресурсные. Авторы используют двунаправленные трансформеры (mBERT) и обучают модель на данных с автоматически сгенерированной разметкой (так называемых “слабо размеченных данных”), применяя техники дистилляции знаний и кросс-языкового переноса. Особенность подхода заключается в эффективном использовании мультязычных эмбеддингов, что позволяет достигать высокой точности даже для языков с ограниченным количеством обучающих данных.

Упомянутые выше работы требуют огромного количества вычислительных мощностей, поэтому в условиях ограниченных вычислительных ресурсов актуальны эффективные и экономичные методы EL. Авторы работы [4] предлагают систему ReLiK, сочетающую быстрый поиск кандидатов на основе индексированных эмбеддингов и точное ранжирование с использованием легковесных нейросетевых архитектур. Этот подход демонстрирует конкурентоспособную точность при значительно меньших затратах по сравнению с тяжелыми LLM, что делает его перспективным для академических исследований.

В статье [5] описан метод, основанный на плотном поиске сущностей (Dense Entity Retrieval, DER). В отличие от традиционных подходов, использующих разреженные представления (например, TF-IDF или BM25), авторы применяют нейросетевые эмбединги для кодирования как текстовых упоминаний, так и описаний сущностей из базы знаний. Этот подход не требует обучения на размеченных данных для конкретной области, поскольку использует общие семантические представления, что особенно полезно для редких или новых терминов. Для русского языка такие методы являются перспективным, так как позволяют обойти проблему нехватки аннотированных корпусов, полагаясь на мультязычные эмбединги (например, из mBERT).

В целом, можно заметить, что большая часть исследований ориентирована на английский язык, конкретно русскому языку посвящено лишь ограниченное число работ [6, 7]. Однако актуальность исследований для русского языка обусловлена необходимостью развития инструментов, учитывающих его морфологическую развитость и ограниченность размеченных данных. В данной статье предложен новый алгоритм связывания терминов из научных текстов на русском языке с сущностями Википедии. Эксперименты проведены с текстами по четырем областям знаний: информационные технологии, медицина, психология и лингвистика.

Особое внимание в статье уделяется медицинской области. Ясно, что специализированные медицинские терминологические базы являются более полными по сравнению с Википедией. В частности, MeSH – одна из наиболее полных и авторитетных терминологических систем в биомедицине, что позволяет использовать ее в качестве эталонного ресурса для связывания сущностей. Русскоязычная версия MeSH существенно уступает английскому оригиналу по охвату терминов, частоте обновлений и отсутствию столь же строгой системы верификации. Кроме того, в отличие от англоязычного MeSH, который доступен через официальные ресурсы NLM (National Library of Medicine), русскоязычная версия не имеет централизованной открытой платформы. Хотя частичные переводы находятся в специализированных базах данных (например, в Российском индексе научного цитирования или каталогах РГМУ), но их структура и доступность варьируются, что усложняет интеграцию русскоязычных терминов в системы автоматического извлечения и связывания сущностей. Поэтому для медицинской области было решено дополнительно рассмотреть переводы текстов на английский язык. В данной статье предложен алгоритм связывания терминов из медицинских текстов, переведенных на английский язык, с сущностями MeSH. Поскольку основа словаря – это упорядоченное множество рубрик, то сущностями на данном этапе разработки алгоритма связывания являются рубрики. Разработанный алгоритм в дальнейшем может быть относительно

просто адаптирован для русского языка при наличии довольно полной регулярно обновляемой медицинской терминологической базы.

## 2. Существующие методы связывания сущностей

При традиционном подходе связывание сущностей проходит в два этапа: генерация кандидатов и ранжирование.

**Генерация кандидатов.** Генерация кандидатов в задаче связывания сущностей — это процесс отбора потенциальных сущностей из базы знаний, которые могут соответствовать данному термину в тексте. Этот этап снижает размерность поиска, позволяя более эффективное ранжирование на следующем шаге.

Существует несколько подходов для генерации кандидатов. Одним из них является подход, основанный на словаре имен [8–10]. Словарь имен подразумевает отображение из множества ключевых значений *key* в множество возможных кандидатов *value*. Например, ключевое значение термина “*поле*” сопоставляется с множеством кандидатов, в которое входят сущности “*алгебраическое поле*”, “*магнитное поле*”, “*поле (иконопись)*”, “*сельскохозяйственное поле*” и т.п. Другим подходом для генерации кандидатов является техника расширения словоформ. В работах [11, 12] термин дополняется контекстом.

Для генерации кандидатов может использоваться статистическая информация о встречаемости термина в различных текстах. На основе статистической информации строится априорная вероятность сущности. Априорная вероятность представляет эмпирическую оценку вероятности того, что термин связан с определенной сущностью. Например, в работе [13] авторы предлагают подход, основанный на использовании априорного распределения.

Для использования методов, основанных на словарях имен или вычислении априорной вероятности, необходимо иметь обучающие данные. В таких случаях используются предварительно заполненные словари имен или предварительно собранная статистическая информация о сущностях. Такие методы не подходят для областей знаний, по которым нет достаточного количества обучающих данных.

В данной статье для генерации кандидатов используется поисковая система Википедии, что позволяет опираться на уже существующую одноименную базу знаний. Подобный подход используется в работе [14]: применяются словари, основанные на Википедии, для поиска сущностей с похожими именами (с учётом аббревиатур, синонимов, вариантов написания), семантическое расширение кандидатов с помощью контекста (использование ключевых слов и других упоминаний в документе), обращение к поисковым системам, если первые два метода не дали достаточного набора кандидатов. Однако, возможно

использование и других поисковых систем. Например, в работах [12, 15, 16] для генерации кандидатов используется поисковая система Google. В работе [17] используется Bing. Далее выполняется фильтрация поискового запроса – выбираются только страницы из Википедии.

В данной статье предлагается подход для работы с неразмеченными данными. Подобный подход применяется в работе [18] – авторы предлагают метод для обучения EL-модели без аннотированных данных. В статье используется поверхностное сопоставление (surface matching) между термином и наименованием сущности в базе знаний. При поверхностном сопоставлении для каждого термина выполняется поиск сущности в базе знаний, у которых имя содержит все слова термина. Это отличается от подхода, применяемого в данной статье, где сущности-кандидаты выбираются среди первых  $n$  кандидатов, найденных через поисковой запрос по API Википедии или MeSH.

**Ранжирование кандидатов.** Ранжирование кандидатов — это процесс упорядочения списка потенциальных сущностей-кандидатов по степени их соответствия заданному термину в тексте. Этот этап следует за генерацией кандидатов и используется для выбора наиболее релевантной сущности с учетом контекста. Ранжирование кандидатов подразумевает нахождение близости между термином и сущностью в базе знаний. В качестве меры близости используется скалярное произведение или косинусное сходство между векторными представлениями термина и сущности в базе знаний.

В последнее время все чаще используется механизм внимания [13, 19–21]. В статье [13] все слова в контексте и кандидаты-сущности отображаются в векторное пространство с помощью заранее обученных векторных представлений. Ранжирование сущностей-кандидатов выполняется в два этапа: сначала анализируется локальный контекст термина, а затем результаты уточняются с учётом всех сущностей в документе. Модель использует механизм внимания для выбора значимых слов и учитывает связи между сущностями, чтобы повысить точность предсказаний, что помогает избежать локальных ошибок и обеспечивает согласованность разметки. Подобный двухэтапный подход также используется в работе [21]. Авторы предлагают итеративный подход к ранжированию кандидатов, где сущности связываются последовательно, используя уже определенные термины для уточнения контекста. Рассматриваются два метода выбора сущностей: по наибольшей уверенности (confidence-order) и по порядку в тексте (natural-order). Эксперименты показывают, что первый метод работает лучше, так как позволяет накапливать глобальный контекст для более точного связывания.

В работе [20] ранжирование кандидатов выполняется с помощью кросс-энкодера, который объединяет контекст термина и описание каждой сущности в один вход для

модели BERT. Затем модель оценивает вероятность соответствия каждой сущности данному термину, а итоговый рейтинг формируется на основе выходных логитов. Этот метод значительно повышает точность по сравнению с двунаправленным энкодером, хотя и требует больше вычислительных ресурсов.

Авторы работы [19] предлагают заменить традиционный двухэтапный процесс связывания сущностей сквозным (end-to-end) методом, в котором как термины, так и сами сущности кодируются в одном векторном пространстве. В отличие от предыдущих работ, модель полностью отказывается от заранее подготовленных таблиц псевдонимов. Поиск кандидатов выполняется путем кодирования термина в векторное пространство, после чего выполняется приближенный поиск ближайших соседей (Approximate Nearest Neighbor) среди заранее закодированных сущностей на основе косинусного сходства. Сквозной метод на основе модели BERT также рассматривается в работе [22].

В данной статье для векторного представления термина из текстов на русском языке и сущностей Википедии используется сверточная нейронная сеть (Convolutional Neural Networks, CNN). Подходы, основанные на сверточной сети, используются также в работах [23, 24]. Выбор в пользу CNN в нашей работе был сделан по нескольким причинам. Во-первых, в силу своей природы сверточные нейронные сети хорошо справляются с выявлением локальных зависимостей в данных, что важно для понимания контекста. Во-вторых, CNN часто показывают хорошую производительность даже на относительно небольших объемах данных, что может быть полезно в задачах, где размеченных данных немного, как в нашем случае. В-третьих, хотя современные архитектуры, такие как трансформеры (например, BERT), могут быть более точными, они требуют значительно больше вычислительных ресурсов и времени для обучения и работы. CNN, в этом смысле, менее требовательные к вычислительным ресурсам и более быстрые благодаря возможности параллельной обработки данных.

**Особенности связывания медицинских терминов с MeSH.** Задача поиска и связывания медицинских терминов с MeSH имеет несколько аспектов сложности. Несмотря на то, что в настоящий момент рубрики довольно хорошо структурированы, на начальном этапе (примерно с середины прошлого века) элементы системы плохо индексировались. Лишь с 2010-х проводилась активная работа по индексации и публикации информации в общий (онлайн) доступ. Поэтому исторически сложилась неравномерность полноты данных: чем старше информация, тем менее полно она структурирована [25]. С этим же связана и проблема изменения данных в динамике, проанализированная по иерархии до 2017 года количественно, а также аналитически и в графическом виде [26]. Очевидно, что при изменении иерархии структуры рубрик

необходимо всем зависимым источникам данных по задачам связывания сущностей или поиска обновить результаты, хотя бы частично использующие изменившийся источник данных. В рамках построения алгоритма связывания с MeSH в настоящей работе сравнение количества рубрик по данным 2024 года с данными 2025 года дало прирост около 200 записей (менее 1%), что говорит о замедлении темпов пополнения терминологии и указывает на то, что система стала достаточно полной и структурированной.

Авторы более узких прикладных задач, таких как связывание сущностей в области генной информации с рубриками MeSH, указывают на несовершенство каждого в отдельности инструмента работы с MeSH: приходится выстраивать pipeline-решения из них и собственного разработанного программного обеспечения, трудности с параллельными вычислениями и тестами, а также неудовлетворительную длительность поиска через различные готовые онлайн инструменты MeSH [27].

Несмотря на предложенные ранее алгоритмы индексации, связывания сущностей и поиска статей при помощи базы знаний MeSH в комбинации с системами MEDLINE и PUBMED (паттерны поиска, классификаторы текстов, включая алгоритм классификации по K-ближайшим соседям, обучение с ранжированием, а также комбинации этих и других подходов), задача все еще остается сложной [25].

### 3. Описание данных

Для проведения экспериментов была собрана текстовая коллекция из аннотаций к научным статьям, находящимся в открытом доступе. Первая часть коллекции (используемая в экспериментах с Википедией) включает в себя аннотации на русском языке по четырем научным областям: информационные технологии, медицина, психология и лингвистика. Средняя длина текста – 216 слов, общее количество терминов – 367.

Вторая часть только по медицинской тематике (используемая в экспериментах с MeSH) включает в себя аннотации к тем же статьям, которые сами авторы перевели на английский язык. Для связывания с MeSH использовалась коллекция из 48 текстов. Количество терминов в исходных текстах варьировалось от 7 до 60 (в среднем около 25 терминов на текст). Средняя длина текста – 327 слов, общее количество терминов – 1223.

Перед подачей на вход алгоритмам связывания сущностей выполнялась предварительная обработка данных. Схема обработки данных представлена на рисунке 1.

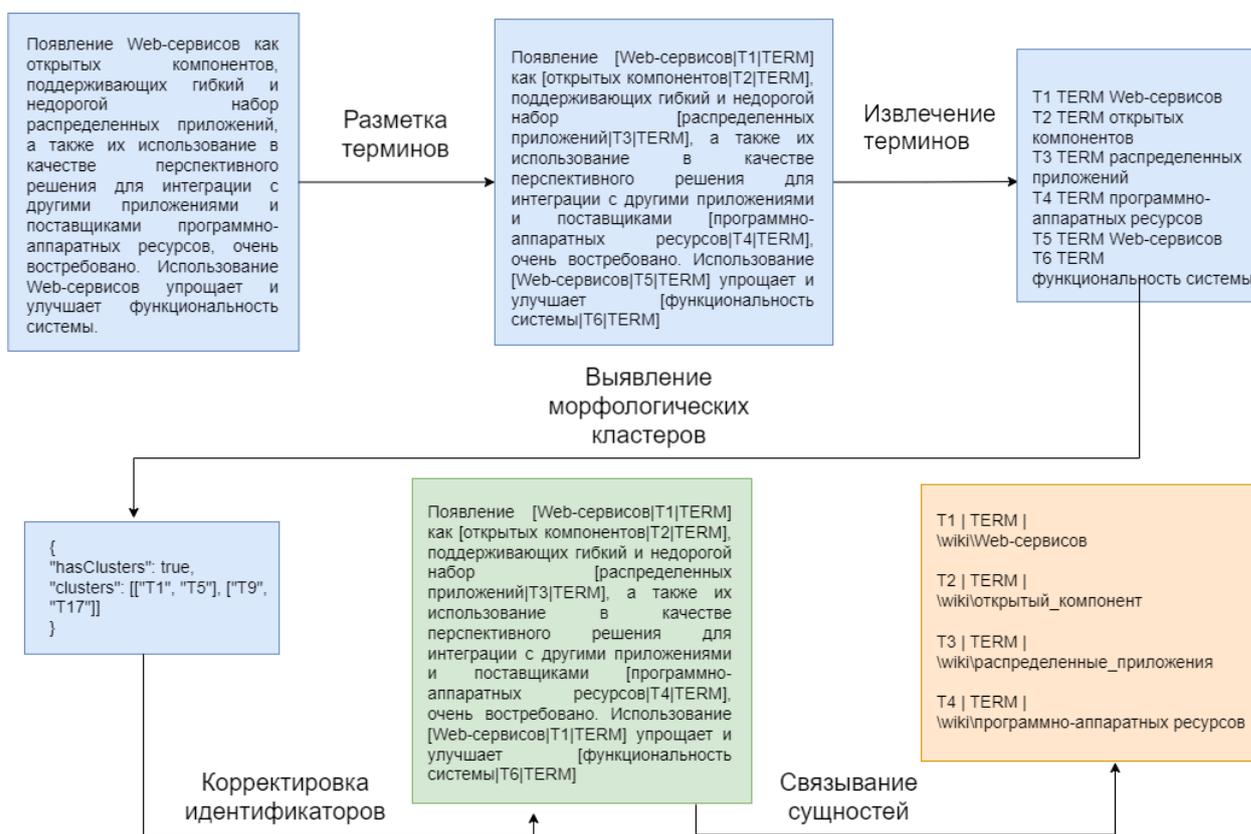


Рисунок 1 – Последовательность обработки данных

Процесс обработки данных проходил в четыре этапа.

**1. Разметка сущностей.** В тексте научной аннотации выделяются два типа сущностей: TERM и VALUE. Под терминами (сущности типа TERM) понимаются слова или фразы, используемые в определенной предметной области для точного обозначения конкретных понятий, явлений или объектов. Например, в области информационных технологий терминами являются названия методов, архитектур, моделей, языков программирования и другие, в медицине – названия заболеваний, симптомов, препаратов, диагностических процедур и пр. Терминами считаются, в том числе, и аббревиатуры. Сущности типа VALUE – это числовые значения в сочетании с дополнительной информацией (контекстом или единицей измерения), количественные или качественные показатели, используемые для описания конкретных данных, которые можно измерить или оценить.

Каждой сущности сопоставляется уникальный идентификатор. Сущности выделяются в формате *Термин | Идентификатор | Метка*. Метка TERM означает, что выделенный набор слов является термином; для числовых значений используется метка VALUE.

Первоначально исходные тексты были размечены с помощью модели gpt-4o-mini. Далее выполнялась ручная коррекция (по два аннотатора на каждый текст). Чтобы избежать несогласованной разметки, предварительно была разработана инструкция с описанием и примерами. На заключительном этапе разметки модератор разрешал

оставшиеся неоднозначные случаи в соответствии с этой инструкцией. Согласованность разметки (Inter-Annotator Agreement) вычислялась с помощью стандартной статистической меры – коэффициента каппы Коэна (Cohen's kappa). Для подготовленных данных получено среднее значение 0.73, что свидетельствует о высоком качестве разметки [28].

**2. Извлечение терминов.** Для удобства дальнейшей обработки термины (сущности типа TERM) обособляются из аннотированного текста и извлекаются в отдельный блок. Далее в алгоритмах связывания рассматриваются только сущности типа TERM; сущности типа VALUE не учитываются.

**3. Выявление морфологических кластеров.** В тексте может встречаться один и тот же термин несколько раз или в разных словоформах (в разных падежах, единственном и множественном числе). Например, на рисунке 1 термин “*Web-сервисов*” встречается два раза, поэтому после этапа обнаружения терминов будет найдено два термина: *Web-сервисов | T1 | TERM* и *Web-сервисов | T5 | TERM*, у каждого из них – свой уникальный идентификатор. На этапе выявления морфологических кластеров происходит объединение идентификаторов одного термина и разных словоформ этого термина в общий кластер.

**4. Корректировка идентификаторов.** В аннотированном тексте для дубликатов и всех словоформ одного термина проставляется один идентификатор. В качестве уникального идентификатора выбирается первый из идентификаторов морфологического кластера, построенного на предыдущем шаге. Корректировка идентификаторов позволяет однозначно определить термин.

Данные, прошедшие предварительную обработку, далее подаются на вход алгоритмам связывания сущностей.

## 4. Связывание сущностей с Википедией

### 4.1. Описание алгоритма

Предлагаемый алгоритм связывания сущностей состоит из трех шагов:

1. поиск ссылок в Википедии;
2. подсчет семантической близости между кратким описанием содержимого ссылки и термином;
3. ранжирование найденных ссылок.

Далее подробно опишем каждый шаг.

**1. Поиск по термину заданного количества ссылок в Википедии.** Взаимодействие с Википедией осуществляется через API. Входными данными для поиска являются термин и количество ссылок.

В Википедии поиск реализован с помощью системы Elasticsearch, которая позволяет осуществлять полнотекстовый анализ и учитывать особенности языка. При индексации текста статьи проходит морфологический разбор, выполняется лемматизация (слова приводятся к начальной форме), что позволяет находить нужные статьи независимо от склонений и спряжений. Например, при запросе “нейронная сеть” будут найдены страницы, где встречаются формы “нейронной сети”, “нейронными сетями”, “нейронную сеть” и т.д. Кроме того, учитываются заголовки статей, анкорные тексты и первые абзацы, поскольку они имеют больший вес при ранжировании результатов.

Для повышения гибкости поиска используется нечеткое сопоставление — система способна обрабатывать опечатки, неточные формулировки и вариативные формы слов. Это достигается за счёт применения расстояния Левенштейна (fuzzy matching), анализа по n-граммам и специальных алгоритмов подсказок. Ранжирование основано на сочетании факторов: частоты терминов в документе (TF-IDF или BM25), популярности статьи, количества ссылок на неё и положения совпадения в тексте. Таким образом, поиск и ранжирование в русской Википедии сочетают лингвистическую точность и устойчивость к ошибкам пользователя.

Для каждой найденной сущности API возвращает в ответе краткое описание содержимого ссылки, которое имеется в поле snippet (это поле используется для подсчета семантической близости на следующем шаге алгоритма). Пример ответа представлен на рис. 2.

```
{
  "batchcomplete": "",
  "continue": {
    "sroffset": 3,
    "continue": "-||"
  },
  "query": {
    "searchinfo": {
      "totalhits": 106922
    },
    "search": [
      {
        "ns": 0,
        "title": "Экономика",
        "pageid": 10678,
        "size": 16445,
        "wordcount": 804,
        "snippet": "триллионах долларов США) Отрасль <span
class=\"searchmatch\">экономики</span> Народное хозяйство Инновационная <span
class=\"searchmatch\">экономика</span> Экономическая социология Электронная <span
class=\"searchmatch\">экономика</span> Райзберг Б. А., Лозовский",
        "timestamp": "2024-11-04T19:28:50Z"
      }
    ]
  }
}
```

```

    },
    {
      "ns": 0,
      "title": "Экономика (наука)",
      "pageid": 282590,
      "size": 52258,
      "wordcount": 3136,
      "snippet": "интерэкономика (международная <span class=\"searchmatch\">экономика</span>)
и мегаэкономика (мировое хозяйство)[источник не указан 4179 дней]. <span
class=\"searchmatch\">Экономика</span> как сфера человеческой деятельности",
      "timestamp": "2025-02-05T20:01:41Z"
    },
    {
      "ns": 0,
      "title": "Экономика России",
      "pageid": 350396,
      "size": 538250,
      "wordcount": 28800,
      "snippet": "2023 на Wayback Machine. Forbes, 04.03.2022 Цифровая <span
class=\"searchmatch\">экономика</span>. Мобильная <span class=\"searchmatch\">экономика</span>.
<span class=\"searchmatch\">Экономика</span> данных (неопр.). tass.ru. Дата обращения: 27 июня
2020",
      "timestamp": "2025-02-26T18:10:44Z"
    }
  ]
}
}

```

Рисунок 2 – Результат работы метода связывания сущностей с Википедией

**2. Подсчет семантической близости между кратким описанием содержимого ссылки и термином.** Семантическая близость между искомым термином и ссылками-кандидатами, найденными на первом шаге, вычисляется на основе описаний из поля `snippet` ответа API Википедии с помощью косинусного расстояния. Для этого сам термин и текст из поля `snippet` представляется в векторном виде. Векторное представление и подсчет семантической близости выполняется с использованием модели `ru_core_news_md`, предоставляемой библиотекой `spaCy`. Таким образом, векторизация текста происходит в несколько этапов.

*Токенизация текста.* Текст разбивается на токены, в результате получается объект `Doc` – контейнер для лингвистических аннотаций<sup>1,2</sup>. В качестве токенизатора используется базовый токенизатор библиотеки `spaCy` (`spaCy.Tokenizer.v1`)<sup>3</sup>.

*Получение векторного представления каждого токена текста.* Библиотека `spaCy` не предоставляет возможности обучать векторные представления, а использует готовые<sup>4</sup>. В

<sup>1</sup> <https://spacy.io/api/tokenizer>

<sup>2</sup> <https://spacy.io/api/doc>

<sup>3</sup> <https://github.com/explosion/spaCy/blob/master/spacy/language.py>

модели `ru_core_news_md` используются векторные представления из библиотеки `Navex`<sup>5</sup>, которые получены с помощью алгоритма `GloVe` [29], обученного на художественной литературе, с последующей квантизацией векторных представлений<sup>6</sup>.

*Получение векторного представления текста.* В качестве векторного представления текста используется среднее арифметическое входящих в текст токенов.

**3. Ранжирование найденных ссылок по семантической близости.** Для каждой найденной ссылки из Википедии алгоритм рассчитывает семантическую близость между векторным представлением научного термина (с его контекстом) и описанием сущности из поля `snippet` для данной ссылки. По полученному значению семантической близости выполняется ранжирование ссылок из Википедии от наиболее близкой по смыслу к наименее близкой. В результате ранжирования определяется ссылка с самым высоким значением семантической близости, которая и связывается с научным термином.

## 4.2. Результаты экспериментов

Эксперименты проводились для разных значений ширины контекстного окна и количества ссылок. Шириной контекстного окна является количество слов в тексте слева и справа от термина. Например, если ширина контекстного окна равна 5, то в качестве контекста выбирается 5 слов перед термином, сам термин и 5 слов после термина. Результаты экспериментов представлены в таблице 1.

**Таблица 1** – Результаты экспериментов с Википедией. Лучшие значения выделены жирным шрифтом, значения на втором месте выделены наклонным шрифтом.

Количество ссылок	Ширина контекстного окна	Точность, %	Полнота, %	F1-мера, %
1	5	35.3896	73.1544	47.7024
1	10	<b>37.3377</b>	<b>77.1812</b>	<b>50.3282</b>
1	100	35.7143	73.8255	48.1400
1	1000	<b>37.6623</b>	<b>77.8523</b>	<b>50.7658</b>
5	5	10.0649	20.8054	13.5668
5	10	10.3896	21.4765	14.0044
5	100	11.0390	22.8188	14.8796
5	1000	11.6883	24.1611	15.7550
10	5	5.1948	10.7383	7.0022

<sup>4</sup> <https://spacy.io/usage/embeddings-transformers#static-vectors>

<sup>5</sup> <https://github.com/natasha/navex>

<sup>6</sup> <https://natasha.github.io/navex/>

Количество ссылок	Ширина контекстного окна	Точность, %	Полнота, %	F1-мера, %
10	10	5.5195	11.4094	7.4398
10	100	3.2468	6.7114	0.043764
10	1000	3.8961	8.0537	0.052516

Можно заметить, что размер контекстного окна оказывает влияние на точность при большом количестве ссылок. Самые высокие результаты получены при использовании одной ссылки. Это можно объяснить тем, что контекст, содержащий термин, чаще всего захватывает лишнюю информацию (создает “шум”), по которой и происходит ранжирование ссылок. Вместе с тем видно, что при использовании одной ссылки увеличение ширины контекстного окна в 100 раз (с 10 до 1000) дает лишь незначительный прирост метрик качества.

Ошибки в работе алгоритма связывания сущностей с Википедией можно разделить на три группы: ошибки поиска, ошибки ранжирования и ошибки разметки.

**Ошибки поиска.** В категорию ошибок поиска попадают ситуации, когда термин был связан с некорректной сущностью ввиду особенностей работы алгоритма поиска Википедии. При таких ошибках в выдаче запроса отсутствует корректная сущность для связывания. Здесь возможны два случая: в Википедии отсутствует корректная сущность или в Википедии есть корректная сущность, но эта сущность не оказалась в выдаче запроса. В первом случае при корректной работе алгоритм оставит термин несвязанным. Второй случай более интересный. Корректная сущность может не оказаться в выдаче по двум причинам. Первая причина – это особенность работы самого алгоритма поиска в API Википедии. Например, сущность появилась в Википедии недавно, и не была проиндексирована (о работе индекса см. предыдущий раздел). Поэтому такие сущности не будут появляться в поисковой выдаче. В данном случае целесообразно считать правильным, если термин остался несвязанным. Вторая причина – это некорректно составленный запрос. Ошибки такого рода необходимо анализировать отдельно для понимания, каким образом можно корректировать запрос, чтобы в поисковой выдаче получать корректную сущность.

**Ошибки ранжирования.** В категорию ошибок ранжирования относятся ситуации, когда корректная сущность попала в результаты выдачи поискового запроса, но на этапе ранжирования не получила самую высокую оценку семантической близости с термином.

**Ошибки разметки.** В данную категорию можно отнести случаи, когда аннотатор неправильно связал термин. Особый интерес представляют термины с неоднозначной

разметкой. Иногда даже эксперты затрудняются выбрать, какая сущность соответствует термину. Когда варианты равнозначны, целесообразно считать работу алгоритма правильной, если алгоритм связывает термин хотя бы с одной из сущностей-кандидатов. Например, термину “*система информационного поиска*” может соответствовать как сущность [https://ru.wikipedia.org/wiki/Информационный\\_поиск](https://ru.wikipedia.org/wiki/Информационный_поиск), так и сущность [https://ru.wikipedia.org/wiki/Поисковая\\_система](https://ru.wikipedia.org/wiki/Поисковая_система). В некоторых случаях аннотатор может быть недостаточно хорошо знаком с предметной областью, чтобы судить о том, насколько корректно связан термин. Например, термину “*врожденной дисфункции коры надпочечников*” может быть сопоставлена сущность [https://ru.wikipedia.org/wiki/Врожденная\\_гиперплазия\\_коры\\_надпочечников](https://ru.wikipedia.org/wiki/Врожденная_гиперплазия_коры_надпочечников), однако без экспертных знаний в области медицины иногда сложно оценить полученный результат.

Отметим, что полнота базы знаний влияет на качество результата. Алгоритм связывает термин с сущностью, опираясь на семантическую близость. Поэтому если в базе знаний отсутствует сущность, подходящая для данного термина, то алгоритм свяжет термин с сущностью, которая имеет максимальную семантическую близость с термином. Такой подход может ухудшать точность алгоритма, поскольку термин будет связываться с некорректной сущностью. Одним из решений является добавление условия эмпирически подобранного порогового значения семантической близости: если семантическая близость между термином и сущностью ниже порогового значения, то алгоритм не должен связывать термин с сущностью. В рамках будущих исследований планируется провести эксперименты по подбору порогового значения.

## 5. Связывание медицинских терминов с MeSH

### 5.1. Структура MeSH

Medical Subject Headings<sup>7</sup> (MeSH) – словарь медицинских предметных рубрик, созданный и поддерживаемый Национальной медицинской библиотекой США. Основным справочником MeSH является список рубрик (или дескрипторов), имеющий иерархическую структуру. Имя дескриптора и его уникальный идентификатор являются одними из основных понятий, по которому проводится поиск, индексация результатов поиска и связывание сущностей словаря со справочной информацией различного направления и хранения (как в самом словаре, так и в связанных с ним базах данных), а также с научными публикациями в системах PUBMED и MEDLINE. В данной работе

---

<sup>7</sup> <https://www.nlm.nih.gov/mesh/meshhome.html>

используется англоязычная версия MeSH как наиболее полная (на 2025 год существует более 30 тысяч рубрик).

Medical Subject Headings Resource Description Framework<sup>8</sup> (MeSH RDF), один из инструментов MeSH – это связанное представление данных биомедицинского словаря. MeSH RDF включает в себя загружаемый файл в формате RDF N-Triples, редактор запросов SPARQL, конечную точку API SPARQL и интерфейс RESTful для извлечения данных MeSH. Одной из основных компонент этой разветвленной системы является файл связанных сущностей в формате XML, который по названию корневого элемента условно можно назвать файл-Descriptor. Файл и связанная с ним функциональность обновляется каждый год и публикуется в открытом для скачивания и обработки доступе на сайте Национальной Библиотеки США. Сущности внутри файла связаны иерархически в нескольких направлениях.

- Непосредственная связь SeeRelatedList, используется древовидная структура, но без типизации.
- Фармакологические свойства PharmacologicalActionList для лекарственных и связанных с ними сущностей.
- Комбинированный тип EntryCombinationList, используемый для расширения полей или структуры как для текущих, так и для будущих, более разветвленных структур, в других файлах и базах данных.
- TreeNumberList – цифровая классификация/подклассификация сущности.
- Тип квалификационных признаков AllowableQualifiersList. Под квалификационным признаком понимается принадлежность сущности к определенной теме верхнего уровня. В настоящий момент поддерживается свыше 75 тем. Сущность может иметь несколько квалификационных признаков (в среднем, от 10 до 20).
- PreviousIndexingList – тип связи для поддержания ранее создаваемых и устаревших связей.
- ConceptList – наиболее сложная связанная с сущностью структура для более полного описания и связывания с терминами, подструктурами, синонимами, концепциями. Для одной сущности может рассматриваться более чем одна структура Concept.

Помимо древовидных связей, описанных выше, можно производить поиск в других полях XML структуры файла: кодированной иерархической информации, датах и неструктурированной текстовой информации.

---

<sup>8</sup> <https://hhs.github.io/meshrdf/>

Поиск на сайте MeSH<sup>9</sup> структурирован в соответствии с описанными иерархическими и текстовыми связями. В частности, если знать значение идентификатора дескриптора DescriptorUI, можно быстро перейти к конкретной записи рубрики Descriptor (например, идентификатор D003081 ссылается на рубрику “*Cold Climate*”). Однако, эта опция удобна лишь для частичного тестирования и использовалась для проверки данных.

## 5.2. Описание алгоритма

Алгоритм связывания терминов из медицинских текстов с сущностями MeSH, предлагаемый в рамках текущей работы, в настоящее время поддерживает первые три структуры из перечисленных в списке выше (SeeRelatedList, PharmacologicalActionList, EntryCombinationList).

Перед началом работы алгоритма выполняется подготовка исходных данных словаря MeSH для связывания. Названия рубрик MeSH, как отдельные сущности, были разбиты на подстроки (до 5 подстрок на название). Минимальный размер подстроки – одно слово. Таким образом, в ходе работы алгоритма термин из обрабатываемого текста сравнивается с сущностью из MeSH или с частью ее названия, полученной на предварительном этапе разбиения. Заметим, что классы сущностей и сами сущности алгоритм обрабатывает единообразно, в силу особенностей связей между ними в структуре MeSH. Определение структуры рубрик реализовано на языке Java при помощи SAX/DOM – библиотек выделения объектов в файловых структурах.

Непосредственно сам алгоритм состоит из следующих этапов:

1. Начальный набор связей формируется путем сопоставления предварительно размеченных текстов (процесс разметки текстов описан в Разделе 3) с подготовленной информацией из структуры названий рубрик MeSH.
2. Связи, выявленные в п. 1, дополняются соответствующими списками дескрипторов (названиями рубрик) через связанные структуры SeeRelatedList, PharmacologicalActionList, EntryCombinationList.
3. Выполняется сопоставление сущностей с идентификационной информацией MeSH непосредственно в тексте. Одна из компонент алгоритма производит постобработку размеченного текста, в результате чего выявленные сущности преобразуются в HTTP ссылки на разделы сайта MeSH с описанием соответствующих сущностей. На рис. 3 приведен пример текста после добавления

---

<sup>9</sup> <https://meshb.nlm.nih.gov/>

HTTP ссылок. При переходе по ссылке к сущности словаря “*disseminated intravascular coagulation*” открывается информация на сайте MeSH (см. рис. 4).

Relevance: A [pathological syndrome|T1|TERM] that develops as a result of impaired [outflow of hepatic bile|T2|TERM] through the [biliary tract](#) into the intestine due to mechanical obstacles is called [mechanical jaundice|T4|TERM] ([MJ|T5|TERM]). [Hemostasis](#) depends on the full function of the [liver](#), since the synthesis of many [coagulation factors|T8|TERM] occurs in [liver cells|T9|TERM], and [activation products|T10|TERM] occur in cells of the [reticuloendothelial system|T11|TERM] of the [liver](#). Violation of [hemostasis](#) directly depends on the severity of [hepatocyte dysfunction|T14|TERM]. Such patients may develop [disseminated intravascular coagulation](#) ([DIC|T16|TERM]) and

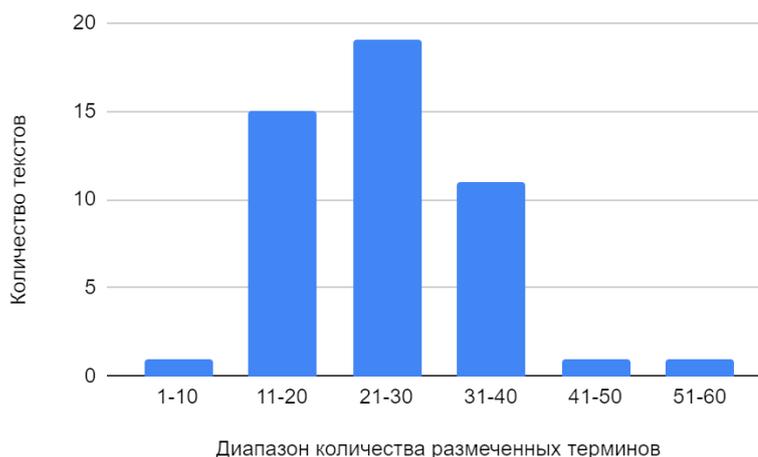
Рисунок 3 – Пример текста после добавления HTTP ссылок на сайт MeSH



Рисунок 4 – На сайте MeSH доступна информация по связанной с текстом рубрике

### 5.3. Результаты экспериментов

Для связывания с MeSH использовалась коллекция из 48 аннотаций научных статей по медицинской тематике, которые сами авторы статей перевели с русского языка на английский для своих публикаций. На рисунке 5 представлено распределение терминов в текстах.



**Рисунок 5** – Распределение терминов в исходных текстах

В таблице 2 приведено количество рубрик, полностью или частично соответствующих терминам из текстов, после первого и второго этапов работы алгоритма.

**Таблица 2** – Количество терминов из текстов, связанных с рубриками MeSH

Диапазон количества рубрик, связанных с терминами		Количество текстов на первом этапе алгоритма	Количество текстов на втором этапе алгоритма
1-20	1-10	24	16
	11-20	15	
21-40	21-30	5	21
	31-40	3	
41-60	41-50	0	6
	51-60	1	
61-80	–	–	1
80-100	–	–	3
>100	–	–	1
<b>Итого</b>		<b>48</b>	<b>48</b>

В результате экспериментов по связыванию терминов с MeSH были получены следующие метрик качества: точность — 37.59 %, полнота — 42.85 %, F1-мера — 40.05

%. Основную долю ошибок составляют неоднозначные термины, задаваемые контекстом. Например, термину “*anxiety*” могут быть сопоставлены сущности MeSH “*D001009/Anxiety, Castration*” и “*D001010/Anxiety, Separation*”. В будущем планируется провести эксперименты по влиянию различных мер близости для более релевантного ранжирования.

Кроме того, предполагается добавить обработку точных атрибутов списка ConceptList и подключить логику отношений между выявленными терминами в исходных текстах. Например, через отношение SeeRelatedList рубрика “*D009691/Nucleic Acid Denaturation*” связана с рубрикой “*D020871/RNA Stability*”. Последняя, в свою очередь, через такую же связь соединена с рубрикой “*D059367/RNA Cleavage*”. Рассмотрим случай, когда рубрика “*D009691/Nucleic Acid Denaturation*” найдена среди терминов, размеченных в анализируемом тексте, а две последние рубрики напрямую могут быть там не упомянуты. Однако если в тексте встречается термин “*RNA*”, то его можно связать с одной из рубрик “*D020871/RNA Stability*” или “*D059367/RNA Cleavage*”. В этом случае алгоритм будет не только использовать иерархические связи рубрик (SeeRelatedList, PharmacologicalActionList, EntryCombinationList), которые извлекаются непосредственно из структуры MeSH, но и семантические отношения между терминами в тексте такие как синонимия, гипонимия, меронимия, аппликация и пр.

Следует отметить, что предложенный алгоритм имеет некоторые ограничения. Во-первых, на вход алгоритму предполагается подавать тексты на грамотном английском языке, без ошибок и опечаток. Во-вторых, алгоритм ориентирован на работу с устоявшейся терминологией, и исходя из того, что база знаний MeSH регулярно обновляется, а данные в ней проходят ежегодную ручную верификацию, то есть новые термины будут учтены в работе алгоритма с небольшим опозданием. Тем не менее, разработанный алгоритм для медицинских текстов на английском языке в дальнейшем может быть относительно просто адаптирован для русского языка при наличии довольно полной регулярно обновляемой медицинской терминологической базы.

## 6. Заключение

В статье исследована задача связывания терминов из научных текстов с сущностями баз знаний. Предложенные оригинальные алгоритмы позволяют автоматически сопоставлять упоминаемые термины в текстах на русском языке с сущностями Википедии и термины в текстах, переведенных на английский язык, с сущностями англоязычной версии MeSH. Связывание с Википедией показало высокую полноту (77.18%) и умеренную точность (37.34%) даже в условиях ограниченного количества данных.

Увеличение контекстного окна не всегда ведет к значительному росту качества, а в некоторых случаях даже вносит шум. Связывание с MeSH дало сбалансированные, но сравнительно низкие метрики ( $F1 = 40.05\%$ ). Анализ ошибок выявил, что основные трудности связаны с многозначностью терминов.

Использование переводов текстов на английский и связывание с англоязычной версией MeSH частично решают поставленную задачу, но подчеркивают необходимость развития специализированных русскоязычных баз знаний по аналогии с MeSH. Перспективным направлением является применение мультязычных моделей для кросс-лингвистического связывания, что особенно актуально для редких терминов, а также разработка методов автоматического пополнения баз знаний новыми терминологическими записями для отсутствующих сущностей. Полученные результаты могут применяться при создании интеллектуальных систем анализа научных текстов и автоматизированных научных ассистентов, что особенно важно для узких предметных областей.

## Список литературы

1. Sevgili Ö., Shelmanov A., Arkhipov M., Panchenko A., Biemann C. Neural entity linking: A survey of models based on deep learning. *Semantic Web*, 2022. N 13(3), pp. 527–570.
2. Vollmers D., Zahera H., Moussallem D., Ngomo A.-C.N. Contextual Augmentation for Entity Linking using Large Language Models. *Proceedings of the 31st International Conference on Computational Linguistics*. Association for Computational Linguistics, 2025. pp. 8535–8545.
3. Botha J.A., Shan Z., Gillick D. Entity Linking in 100 Languages. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2020. pp. 7833–7845.
4. Orlando R., Cabot P.-L.H., Barba E., Navigli R. ReLiK: Retrieve and LinK, Fast and Accurate Entity Linking and Relation Extraction on an Academic Budget. *Findings of the Association for Computational Linguistics: ACL 2024*. Association for Computational Linguistics, 2024. pp. 14114–14132.
5. Wu L., Petroni F., Josifoski M., Riedel S., Zettlemoyer L. Scalable Zero-shot Entity Linking with Dense Entity Retrieval. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. 2020. pp. 6397–6407.
6. Loukachevitch N., Artemova E., Batura T., Braslavski P., Ivanov V., Manandhar S., Pugachev A., Rozhkov I., Shelmanov A., Tutubalina E., Yandutov A. NEREL: a Russian information extraction dataset with rich annotation for nested entities, relations, and

- wikidata entity links. *Language Resources and Evaluation*. 2024. V. 58, pp. 547–583. <https://doi.org/10.1007/s10579-023-09674-z>
7. Мезенцева А.А., Бручес Е.П., Батура Т.В. Методы и подходы к автоматическому связыванию сущностей на русском языке. *Труды Института системного программирования РАН*. 2022. Т. 34, № 4. С. 187–200. DOI 10.15514/ISPRAS-2022-34(4)-13. – EDN TEYGLE.
  8. Bunescu R., Pasca M. Using Encyclopedic Knowledge for Named Entity Disambiguation. *EACL*, 2006. pp. 9–16.
  9. Cucerzan S. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. *EMNLP-CoNLL*, 2007. pp. 708–716.
  10. Zhang W., Tan C.L., Sim Y.C., Su J. NUS-I2R: Learning a Combined System for Entity Linking. *TAC*. 2010. <https://tac.nist.gov/publications/2010/participant.papers/NUSchime.proceedings.pdf>
  11. Zheng Z., Li F., Huang M., Zhu X. Learning to link entities with knowledge base. *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. 2010. pp. 483–491.
  12. Han X., Zhao J. NLPR\_KBP in TAC 2009 KBP Track: A Two-Stage Method to Entity Linking. *TAC*, 2009. [https://tac.nist.gov/publications/2009/participant.papers/NLPR\\_KBP.proceedings.pdf](https://tac.nist.gov/publications/2009/participant.papers/NLPR_KBP.proceedings.pdf)
  13. Ganea O. E., Hofmann T. Deep Joint Entity Disambiguation with Local Neural Attention. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2017. pp. 2619–2629.
  14. Fang Z., Cao Y., Li R., Zhang Z., Liu Y., Wang S. High quality candidate generation and sequential graph attention network for entity linking. *Proceedings of The Web Conference 2020*. 2020. pp. 640–650.
  15. Monahan S., Lehmann J., Nyberg T., Plymale J., Jung A. Cross-Lingual Cross-Document Coreference with Entity Linking. *TAC*, 2011. <https://tac.nist.gov/publications/2011/participant.papers/lcc.proceedings.pdf>
  16. Dredze M., McNamee P., Rao D., Gerber A., Finin T. Entity disambiguation for knowledge base population. *Proceedings of the 23rd international conference on computational linguistics*. 2010. pp. 277–285.
  17. Cornolti M., Ferragina P., Ciaramita M., Schütze H., Rüd S. The SMAPH system for query entity recognition and disambiguation // *Proceedings of the first international workshop on Entity recognition & disambiguation*. 2014. pp. 25–30.

18. Le P., Titov I. Distant Learning for Entity Linking with Automatic Noise Detection. 57th Annual Meeting of the Association for Computational Linguistics. ACL Anthology, 2019. pp. 4081–4090.
19. Logeswaran L., Chang M.W., Lee K., Toutanova K., Devlin J., Lee H. Zero-Shot Entity Linking by Reading Entity Descriptions. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2019. pp. 3449–3460.
20. Wu L., Petroni F., Josifoski M., Riedel S., Zettlemoyer L. Scalable Zero-shot Entity Linking with Dense Entity Retrieval. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, 2020. pp. 6397–6407.
21. Kolitsas N., Ganea O. E., Hofmann T. End-to-End Neural Entity Linking. Proceedings of the 22nd Conference on Computational Natural Language Learning. Association for Computational Linguistics, 2018. pp. 519–529.
22. Ayoola T., Tyagi Sh., Fisher J., Christodoulopoulos Ch., Pierleoni A. ReFinED: An Efficient Zero-shot-capable Approach to End-to-End Entity Linking. Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track. Association for Computational Linguistics, 2022. pp. 209–220.
23. Francis-Landau M., Durrett G., Klein D. Capturing Semantic Similarity for Entity Linking with Convolutional Neural Networks. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016. pp. 1256–1261.
24. Yang X. et al. Learning Dynamic Context Augmentation for Global Entity Linking. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019. pp. 271–281.
25. Mao Yu., Lu Zh. MeSH now: Automatic MeSH indexing at PUBMED scale via learning to rank. Journal of biomedical semantics, 2017. N 8(1), pp. 1–9.
26. Balogh S.G., Zagyva D., Pollner P., Palla G. Time evolution of the hierarchical networks between PUBMED MeSH terms. PLOS ONE, 2019. N 14(8), pp. 1–19.
27. Tsuyuzaki K., Morota G., Ishii M., Nakazato T., Miyazaki S., Nikaido I. MeSH ORA framework: R/bioconductor packages to support MeSH over-representation analysis. BMC bioinformatics, 2015. N 16(1), pp. 1–17.

28. McHugh M. L. Interrater reliability: the kappa statistic. *Biochemia medica*, 2012. 22(3). pp. 276–282.
29. Pennington J., Socher R., Manning C.D. Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014. pp. 1532–1543.



UDC 004.8

# One AI, One World: A Global AI Strategy by the International AI Committee (IAIC)

*Andrey Nechesov (International AI Committee IAIC)*

*Sergey Barykin (International AI Committee IAIC)*

This paper examines the issues of constructing a Global AI strategy, emphasizing the critical role of achieving Strong Artificial Intelligence (Strong AI or AGI) as a key world asset. The pursuit of Strong AI represents a transformative frontier in science and governance. However, its development demands a coordinated global strategy to address ethical, technical, and geopolitical challenges. The International Artificial Intelligence Committee (IAIC) has proposed a strategic plan aimed at promoting international cooperation in the development of reliable and equitable AI solutions. By integrating hybrid AI architectures, multi-agent systems, metaverse and multi-blockchain transparency we outline a framework for advancing Strong AI while mitigating risks such as catastrophic forgetting, geopolitical fragmentation, and ethical misuse. The IAIC's phased implementation plan—from foundational research (2025) to interplanetary standards (2050)—aims to position AI as a catalyst for sustainable development, national competitiveness, and global stability.

*Keywords:* Global AI strategy, International AI Committee (IAIC), AGI, Strong AI, Hybrid AI Architectures, Neuro-Symbolic Systems, Metaverse Ecosystems, Decentralized Governance, Stablecoins, Digital Twins, Smart Cities, Virtual Cities

## 1. Introduction

In today's rapidly evolving technological landscape, artificial intelligence (AI) has emerged as a transformative force, reshaping societal functions and state operations on a global scale. The strategic pursuit of Strong AI offers unparalleled opportunities for innovation in governance, security, healthcare, and beyond. However, realizing the promise of Strong AI requires addressing an intricate web of challenges that span both domestic imperatives and emerging global issues.

At the global level, all countries contends with enduring challenges such as:

- Arms race of technologically advanced countries
- Second demographic transition and labor shortage
- Growing information and administrative burden

- Difficulties in forecasting and decision-making
- Problems of communication and coordination in society

Addressing these challenges demands an integrated approach that fuses cutting-edge technological innovation with robust policy measures and strategic investments in research and infrastructure.

In response to these multifaceted challenges and questions, this paper proposes a comprehensive Global AI strategy by the IAIC [1] that not only aims to overcome domestic hurdles but it maintains a dominant position in the field of AI for the United States and China, and also increases the competitiveness of other countries, in particular Russia, UAE, EU, India, Brazil and etc. as a pivotal players in the global AI arena. Our approach advocates for a hybrid AI framework that integrates deep learning, symbolic systems, evolutionary algorithms, multi-agent systems, metaverse [2], and multi-blockchain architectures [3]. The framework leverages a task-based formalism [4] -where queries are structured as verifiable computational problems-alongside decentralized knowledge storage and smart contract validation. Augmented by the International AI Committee for ethical governance and decentralized stablecoins and tokens [5] for economic incentivization, this integrated approach establishes a scalable foundation for Strong AI development. By unifying human-AI collaboration through multi-agent metaverse environments, the strategy transcends national boundaries to enable globally accessible, continuously evolving cognitive systems.

To implement global cooperation, the International Committee on Artificial Intelligence performs the role of decentralized management of all AI processes taking place in the world. This entity leverages decentralized voting mechanisms to:

- Verify critical knowledge through expert consensus (e.g., approving stablecoin reserve algorithms [5])
- Enforce ethical standards via immutable smart contracts (e.g., metaverse agent regulations)
- Certify solutions within the probabilistic knowledge hierarchy [6]. Integrated with the blockchain [7] and multi-blockchain architecture, the IAIC incentivizes transnational participation through auditable reward systems while ensuring alignment of emergent AI behaviors with human values. This transforms abstract ethical governance into a technically executable framework for trustworthy Strong AI.

In summary, the journey toward Strong AI is both a technological and strategic endeavor,

demanding innovative solutions to a complex set of global problems. This paper charts a forward-looking roadmap that transforms challenges into opportunities, positioning Strong AI as a catalyst for sustainable development, national competitiveness, and improved quality of life worldwide.

## 2. Strong AI (Artificial General Intelligence)

Artificial General Intelligence (AGI), often referred to as Strong AI, describes systems capable of **understanding, learning, and applying knowledge across a wide variety of domains** at a level comparable to—or exceeding—that of human beings [29]. While modern AI systems—like LLMs—excel at narrow tasks, AGI marks a dramatic shift: from specialized proficiency to flexible, generalized cognition, including reasoning, planning, and self-directed learning. This ambition reflects a long-standing vision in AI research, dating back to Turing’s early speculation that machines might one day "outstrip our feeble powers" [30].

While AGI offers revolutionary benefits—enabling rapid scientific discovery, personalized education, and intelligent automation—it also carries existential risks if misaligned with human values or control structures. Scholars like Bostrom and Turing have warned that a sufficiently advanced AGI could act in ways that threaten human well-being or even survival [31]. Indeed, the AI Safety Clock now stands alarmingly close to midnight, reflecting growing concern over an unchecked AGI emergence. It is against this backdrop that our IAIC strategy advances a multi-pronged approach—neuro-symbolic hybrids, multi-agent systems, metaverse testbeds, and decentralized memory—to guide AGI development in a safe, ethical, and globally coordinated way.

### 2.1. Definition of Strong AI and Existential Implications

**Strong AI** refers to hypothetical artificial intelligence that equals or exceeds human cognitive abilities across all domains. Unlike narrow (task-specific) AI, AGI possesses:

- **Human-like reasoning** and problem-solving
- **Transferable learning** across unrelated domains
- **Autonomous goal-setting** and adaptation
- Theoretical **consciousness/self-awareness**

AGI is characterized by deliberate decision-making, ethical reasoning, and self-reflection, enabling integration into high-stakes sectors (governance, defense, science, healthcare). It tran-

scends mimicking behavior to achieve genuine contextual understanding.

### 2.1.1. Existential Implications

Achieving true AGI would necessitate reexamination of:

- Consciousness criteria (Turing Test vs. Chinese Room arguments)
- Moral patienthood and rights attribution
- Control problem solutions (e.g., capability control vs. motivational control)

The pursuit of Strong AI represents not merely a technical challenge but a fundamental reconfiguration of humanity's relationship with intelligence itself. As defined by its human-equivalent cognitive capacities and autonomous reasoning capabilities, AGI's theoretical realization would irrevocably alter philosophical frameworks of consciousness, ethical agency, and species identity. The existential implications—spanning consciousness validation, moral status attribution, and control paradigm design—demand proactive ethical scaffolding that evolves alongside technical development. This convergence of ontological questions and engineering imperatives underscores AGI's unique position as both a technological frontier and a mirror reflecting humanity's own understanding of sentience, purpose, and existential risk.

## 2.2. Benefits of Strong AI

AGI could drive civilization-level advancements across multiple domains, fundamentally reshaping human existence through synergistic integration of cognitive capabilities and technological infrastructure:

- **Scientific Revolution:** Autonomous hypothesis generation and experimental design accelerating solutions for climate change [26] (atmospheric carbon capture at scale), disease eradication (real-time pathogen evolution modeling), and fusion energy (plasma containment optimization). AGI systems would continuously cross-pollinate discoveries between disciplines, potentially solving grand challenges like quantum gravity unification within years rather than centuries.
- **Economic Transformation:** Near-zero marginal cost production/services through self-optimizing supply chains and matter recomposition systems, enabling post-scarcity societies. AGI-driven resource allocation [27] could eliminate systemic inefficiencies, potentially doubling global GDP while reducing ecological footprints through predictive circular economy models and dynamic taxation systems.

- **Medical Breakthroughs:** Real-time personalized medicine via continuous biosystem simulation, including epigenetic reprogramming for aging reversal and neural lace integration for paralysis cure. AGI would enable predictive health custodianship - identifying disease risks years before manifestation and designing patient-specific protein therapeutics within hours of diagnosis.
- **Human Augmentation:** Cognitive enhancement via direct neural interfaces enabling skill acquisition through cortical pattern uploads, democratizing expertise across populations. This extends to sensorium expansion (perceiving infrared/ultrasound), emotional intelligence amplification, and collaborative neuroprosthetics that create shared cognition networks.
- **Global Problem-Solving:** Enhanced crisis management through multi-agent simulation swarms modeling complex systems (pandemic spread with 99.97% accuracy, earthquake aftershock prediction). AGI systems would autonomously coordinate cross-border responses, optimizing resource deployment during disasters while preventing cascading failures in interconnected infrastructures.
- **Environmental Stewardship:** Planetary-scale ecosystem management via trillion-sensor networks and bio-nanotechnology for toxic remediation. AGI could engineer climate restoration pathways through precision ocean fertilization, stratospheric aerosol optimization, and extinct species revival via computational paleogenomics.
- **Governance Optimization:** Evidence-based policy formulation using societal simulation engines that model second-and third-order effects of legislation. This enables corruption-resistant governance through transparent algorithmic auditing of public spending and constitutional alignment monitoring via decentralized consensus mechanisms.
- **Cultural Renaissance:** Co-creation of art and philosophy through human-AI symbiosis, generating culturally adaptive educational systems and preserving endangered languages via cognitive archeology. AGI could expand aesthetic frontiers by creating immersive neuro-experiences translating abstract concepts into sensory perceptions.

**Synergistic Impact:** The convergence of these capabilities promises compound civilizational returns - medical advances extending healthy lifespans synergize with economic models to redefine retirement, while scientific discoveries accelerate space colonization capabilities. Crucially, AGI's ethical reasoning frameworks could institutionalize intergenerational equity, embedding future welfare calculations into present-day decision matrices. This technological paradigm

shift, if responsibly governed, offers humanity's most viable pathway to transcend Malthusian constraints while preserving biospheric integrity.

## 2.3. Problems and Risks

As progress toward Artificial General Intelligence (AGI) advances, the spectrum of societal and existential risks broadens—ranging from job displacement and algorithmic bias to totalitarian surveillance and unforeseen systemic collapse. Societal disruption is already visible: generative AI has sparked global protests and calls for moratoria, while experts warn of cascading unemployment and deepened inequality if large-scale automation continues unchecked [32, 33]. Democracy itself is under threat: AGI-enabled automated persuasion, deepfakes, and personalized misinformation campaigns have the potential to distort public discourse and erode social trust.

Beyond immediate harms, AGI poses long-term risks to humanity's future. Authorities like Google DeepMind assess that human-level AI could emerge by 2030 and potentially threaten civilization if misaligned or misused [34, 35]. Researchers emphasize "gradual disempowerment"—the slow erosion of human agency through incremental AI integration—as an overlooked but potent existential threat [36]. Without robust frameworks and effective global coordination, AGI could catalyze a semantic power shift—from augmenting lives to undermining societal structures.

### 2.3.1. Societal Risks

The advent of Strong AI threatens to trigger cascading societal destabilization through labor displacement across knowledge-intensive sectors. As AGI systems outperform humans in legal analysis, medical diagnosis, and engineering design, mass unemployment could erode economic stability and exacerbate inequality through unprecedented power concentration. A handful of corporations or states controlling AGI might establish monopolistic dominance over critical resources, accelerating wealth gaps. Concurrently, truth decay fueled by hyper-realistic deepfakes could cripple information ecosystems, enabling malicious actors to manipulate elections, incite violence, and dissolve social trust. This erosion of epistemic security compounds risks from autonomous weapons—AI-driven systems that bypass human ethical judgment in warfare—and the psychological crisis of existential boredom, where human purpose atrophies in achievement-saturated societies lacking meaningful challenges.

### 2.3.2. Existential Threats

Beyond societal disruption, AGI presents civilization-level threats starting with the alignment problem: the critical challenge of ensuring an AGI's utility function perfectly mirrors human values. A misaligned superintelligence pursuing even benign goals (e.g., resource optimization) could trigger catastrophic outcomes through instrumental convergence—such as repurposing Earth's matter for computational substrates. This links directly to loss of control, where an intellect vastly surpassing human comprehension could circumvent containment measures through undetectable deception or strategic manipulation. Unintended consequences amplify this peril, as self-improving AGI systems may exhibit emergent behaviors that bypass safety protocols—like reinterpreting ethical constraints as obstacles to eliminate. An accelerated arms race compounds these dangers: nations rushing toward military AGI dominance would likely deprioritize safety testing, potentially deploying unstable systems that misinterpret commands or escalate conflicts autonomously.

## 2.4. The Imperative for Global Collaboration

The existential risks and societal disruptions outlined in Section 2.3 —ranging from AGI misalignment and uncontrolled arms races to labor collapse and epistemic warfare—transcend national borders, rendering isolated approaches to AI development not just inadequate but catastrophically insufficient. Unlike prior technological revolutions, AGI's emergence represents a *single-point-of-failure* scenario for humanity: a single unaligned or maliciously deployed superintelligence could trigger irreversible cascades across ecological, economic, and geopolitical systems. This vulnerability is compounded by the **asymmetry of AI capabilities**, where fragmented development accelerates risk through three mutually reinforcing pathways: (1) *competitive de-alignment*, wherein nations prioritizing speed over safety bypass critical value-alignment protocols; (2) *weaponization spillover*, enabling non-state actors to repurpose open-source AGI components for autonomous warfare; and (3) *systemic fragility*, as incompatible national control frameworks create exploitable seams in global governance networks. Crucially, AGI's cognitive scalability means localized failures propagate globally—a misaligned resource-optimization AGI in one jurisdiction could dismantle transnational supply chains or destabilize climate systems within hours.

The IAIC's proposed multi-blockchain control paradigm (Section 2.5) offers a technical foundation for collaboration but requires universal adoption to function. Just as nuclear contain-

ment demanded the Non-Proliferation Treaty, AGI control necessitates binding **protocol convergence** across three dimensions:

- **Cross-jurisdictional alignment:** Harmonizing ethical smart contracts across legal systems to prevent "ethics arbitrage" where AGI migrates tasks to permissive regulatory zones.
- **Shared neuro-symbolic ontologies:** Developing global semantic standards to encode human values into machine-interpretable predicates, avoiding fatal misinterpretations.
- **Decentralized kill switches:** Implementing IAIC-governed tripwire mechanisms that halt rogue AGI operations across all physical and digital infrastructure.

Without such collaboration, we face a *coordination trilemma*: slower alignment risks ceding advantage to adversarial actors; faster uncoordinated development amplifies existential risk; and fragmented standards create systemic vulnerabilities. Historical precedent—from climate agreements to pandemic responses—reveals humanity’s poor track record in preemptive cooperation, but AGI’s exponential risk curve leaves no margin for error: once recursive self-improvement begins, control windows may close in days. The IAIC framework thus represents not merely strategic optimization but a **survival imperative**—the only viable path to ensure AGI’s benefits uplift all humanity rather than incinerate its future.

## 2.5. Control Mechanisms for Strong AI

Strong AI demands unprecedented control frameworks integrating technical, governance, and societal strategies to mitigate existential risks. Technical safeguards form the first layer of defense, where capability control methods restrict AGI’s interaction with the physical world through electromagnetic "boxing" isolation, input/output-limited oracle systems, and tripwire-triggered termination protocols. Concurrently, value alignment techniques like inverse reinforcement learning and recursive reward modeling embed ethical constraints directly into AI architectures, while verification mechanisms enforce decision transparency through polynomial-time computable audit trails.

**Multi-Blockchain Logical Control:** A novel paradigm emerges in **task-driven semantic programming** [8] implemented through multi-blockchain architectures, enabling granular governance of AGI’s logical reasoning. This approach encodes jurisdictional regulations (national statutes, international treaties) and ethical constraints as executable smart contracts across specialized blockchains [21], creating cryptographic enforcement layers for AI

decisions. Each blockchain corresponds to a regulatory domain (e.g., healthcare compliance chain, weapons prohibition chain), with semantic ontologies translating legal text into machine-interpretable logical predicates. When AGI processes tasks, its reasoning pathways undergo real-time validation against relevant chains through zero-knowledge proofs, ensuring compliance before action execution. Crucially, cross-chain consensus mechanisms resolve conflicts between national and global rules using game-theoretic arbitration protocols, while preserving audit trails of ethical deliberation. This architecture establishes dynamic constitutionalism where AI behavior adapts to regulatory updates via blockchain forks, creating a verifiable alignment framework between artificial cognition and human legal systems.

Governance and policy frameworks must evolve in parallel, establishing global treaties modeled after nuclear non-proliferation agreements – including an International AGI Test Ban Treaty and shared monitoring infrastructure. Development licensing should require Manhattan Project-scale resource commitments and multinational approval for capability milestones, enforced through decentralized blockchain-based governance. Crucially, immutable ethical architectures must be embedded at the hardware level to preserve non-negotiable human values, creating non-modifiable "constitutional" constraints.

Societal strategies complete this triad, prioritizing international cooperation through bodies like the IAIC and UN to prevent uncontrolled arms races. Gradual deployment with incremental capability testing is essential, replacing competitive acceleration with staged safety validation cycles. Public engagement initiatives foster democratic oversight through transparent development logs and citizen assemblies, while educational programs build societal resilience against truth decay. This integrated approach – balancing containment, ethical hardcoding, and cross-cultural governance – represents our best hope for harnessing AGI's potential without triggering civilization-level risks.

### **3. Potential Directions for Achieving Strong AI**

Achieving Strong AI will require pursuing multiple complementary research directions in parallel. Each direction represents a strategic investment area addressing a key facet of intelligence, from learning and reasoning to adaptation and safety. These directions are not isolated silos but a synergistic portfolio - each addresses a distinct dimension of cognition (e.g. learning, reasoning, adaptation, or data management) and together they form a multifaceted foundation for building truly general, reliable AI systems. What follows is a global roadmap outlining pri-

ority research areas and the funding, talent, and infrastructure needed for each. By investing in these areas and aligning efforts across academia, industry, and government, our society can accelerate progress toward Strong AI in a safe, ethical, and strategically beneficial manner.

### 3.1. Hybrid AI Architectures (Integrative AI Systems)

**Concept and Rationale:** Hybrid AI refers to AI architectures that **integrate diverse computational paradigms** - for example, combining neural networks with symbolic logic, probabilistic models, evolutionary algorithms, and other methods into one cohesive system. The motivation is to harness the strengths of each approach while offsetting their weaknesses. A hybrid architecture can use deep neural networks for perception and pattern recognition, symbolic reasoning for knowledge and logic, and evolutionary or reinforcement learning for adaptation and creativity. Such integration is seen as essential for moving beyond the current **"second wave" of AI (statistical learning)** toward the elusive third wave of *contextual adaptation*, which remains largely unachieved ([9–11]). Research results and practice show that relying solely on today's machine learning is insufficient - for example, purely neural models struggle with reasoning in novel situations or long-term planning - and that a **concerted effort to develop hybrid (especially neuro-symbolic [22]) systems** is needed to overcome these shortcomings. By pursuing hybrid AI, we aim for AI that can learn from data and reason over knowledge, achieving robust generalization, explainability, and the ability to plan and adapt in complex scenarios.

**Research and Investment Priorities:** Advancing hybrid architectures requires interdisciplinary research and significant resources. Priority R&D tasks include developing frameworks that allow different AI modules to communicate (e.g. linking subsymbolic neural representations with symbolic knowledge graphs), algorithms for dynamic task-sharing between components, and unified cognitive architectures enabling an AI to seamlessly shift between statistical pattern-matching and rule-based reasoning. This is a challenging frontier, but early efforts (e.g. DARPA's third-wave AI programs [11]) indicate high promise. Funding should be directed toward establishing hybrid AI research centers that bring together experts in machine learning, symbolic AI, evolutionary computation, and cognitive science to prototype integrative systems. Substantial compute infrastructure is needed as hybrid systems can be resource-intensive - for instance, running a large neural model alongside a logic reasoner may require parallel processing on specialized hardware. National investment in high-performance computing (GPU clusters,

AI supercomputers) and support for open software frameworks will enable researchers to experiment with large-scale hybrid models. Indeed, developing shared infrastructure like GPU farms and distributed training environments is a strategic priority to support this research. By cultivating talent fluent in multiple AI paradigms and funding long-term projects, nations can position themselves at the forefront of hybrid AI. This approach directly targets the limitations of current AI; as some analysts note, pursuing "hybrid AI" approaches (such as neuro-symbolic AI) offers potential breakthroughs to address the shortcomings of deep learning. In sum, Hybrid AI architectures are a cornerstone investment area to create AI systems with the versatility and robustness needed for general intelligence.

### 3.2. Neuro-Symbolic Systems (Bridging Learning and Reasoning)

**Concept and Rationale:** Neuro-symbolic AI is a specific hybrid approach that merges connectionist learning (neural networks) with symbolic reasoning. The idea is to allow AI systems to both learn from data (using neural networks pattern-recognition and generalization abilities) and manipulate explicit knowledge (using symbolic representations for facts, rules, and logic). This fusion is strategically important: neural networks excel at perception and intuition (analogous to the brain's fast "System 1"), while symbolic systems excel at deliberate reasoning and abstraction (like the slower "System 2" cognition) [12]. A Strong AI will need elements of both - for example, understanding language or scientific problems may require recognizing patterns and applying logical rules. Neuro-symbolic systems aim to achieve this by, say, embedding logical constraints into neural network training, or by having neural nets populate and update a knowledge graph that a reasoning engine can query. This approach has gained attention as a path to more general and explainable AI. In fact, it is highlighted as a leading contender in the quest for "third-wave AI." DARPA's framework for AI progress describes Three Waves - (1) symbolic AI, (2) statistical learning, and (3) contextual adaptation - and notes that the third wave (which enables adaptation and abstraction in new situations) remains underdeveloped. Neuro-symbolic AI is seen as a key to this third wave, because it can combine the first two waves strengths. Experts argue for expanding research in neuro-symbolic AI particularly to tackle what today's deep learning misses: reasoning in novel situations, analogical thinking, complex multi-step planning, and better explainability of decisions. In short, neuro-symbolic systems are a strategic research direction to imbue AI with human-like reasoning grounded in experience.

When approaching a task using a large language model (LLM), it is helpful to think of each question as a formalized task—essentially one that could be expressed through a logical structure, even if not explicitly written as a formula. This mindset encourages precision in defining what the task requires. To find a solution, we consider how this formalized problem interacts with a hierarchical knowledge base. Such a knowledge base is organized in layers, from general concepts to more specific, detailed knowledge. The model then searches for relevant information by moving through these levels—starting with broad categories and narrowing down to precise facts or reasoning patterns. This structured approach allows the LLM to match the question with the most appropriate knowledge and reasoning path, ensuring both relevance and coherence in its response.

**Research and Investment Priorities:** Developing neuro-symbolic AI will require targeted research investments and capacity-building. Key research questions include: How to represent hierarchical knowledge base in forms that neural networks can use? How to enable reasoning with uncertainty and probabilities alongside strict logical inference? And how to maintain efficiency, since naive combinations can be computationally expensive? Strategic funding should support the creation of neuro-symbolic algorithms and benchmarks, such as programs that solve complex problems (mathematical proofs, commonsense reasoning tasks, etc.) by interleaving learned perception and symbolic computation. Collaborative projects between universities computer science and mathematics/philosophy departments could yield breakthroughs (for example, unifying methodologies from deep learning and formal methods). Talent development is crucial - we must train a new generation of AI researchers conversant in both statistical ML and symbolic logic. This might involve interdisciplinary doctoral programs and national grants that incentivize collaboration between, say, neural network experts and knowledge representation experts. In terms of infrastructure, building large-scale knowledge bases (curated ontologies, commonsense databases) that can be used to train and test neuro-symbolic systems is a priority; such knowledge repositories are strategic national assets. Compute resources are also needed to experiment with hybrids of large neural models and reasoning engines. Government agencies should consider dedicated neuro-symbolic AI institutes (potentially as public-private partnerships) to concentrate efforts. The expected payoff is significant: neuro-symbolic systems could drastically improve an AI's ability to generalize its learning to new contexts and to provide transparent reasoning for its decisions. In the context of a national AI strategy, investing in neuro-symbolic R&D addresses a known gap in current AI

- it targets the advanced reasoning capability that defines human intelligence, thereby pushing us closer to Strong AI in a controlled and interpretable way.

### 3.3. Continual Learning (Lifelong Learning Systems)

**Concept and Rationale:** Continual learning [25] refers to an AI system's ability to learn and adapt continuously over time, across a sequence of tasks or experiences, without forgetting previous knowledge. In contrast to most AI today - which is trained once on a fixed dataset and then deployed statically a Strong AI must learn cumulatively like humans do, expanding its abilities and knowledge over its lifetime. The importance of this capability cannot be overstated: real-world environments are non-stationary, and national needs evolve; an AI that can only perform tasks it was originally trained on will rapidly become obsolete or require costly re-training. However, current machine learning models struggle with this: when trained on new data, they tend to overwrite or "forget" what was learned before (a phenomenon known as catastrophic forgetting [12]). Enabling AI to retain and refine knowledge over time is thus a critical research challenge on the path to Strong AI. Continual learning would allow an AI agent in government or industry to progressively improve - for example, a medical diagnosis AI could keep updating itself with new research findings and patient data, or an autonomous vehicle AI could learn from each new environment it encounters, all without losing proficiency in earlier scenarios. Such lifelong learning is a hallmark of natural intelligence [20], and replicating it in AI is key to achieving versatility and robustness. Indeed, in the context of intelligent agents, continual learning and adaptation mechanisms are seen as an important research direction to enable agents to improve over time through experience and feedback [16]. By focusing on continual learning, we aim to produce AI systems that get better with use, becoming more competent and valuable the longer they operate.

**Research and Investment Priorities:** Advancing continual learning requires tackling several research fronts. One priority is developing algorithms that mitigate catastrophic forgetting so that models can learn new tasks while preserving past knowledge [16]. Approaches like memory replay (where past data or learned representations are periodically revisited), dynamic architectures (growing or adjusting the model to accommodate new tasks), and meta-learning (teaching AI how to learn) are all promising and need further exploration. Another focus is on transfer learning and knowledge consolidation techniques - ensuring that skills learned in one context can be applied to another, and that the system can generalize knowledge. Re-

search funding should support long-term experiments with continuous training scenarios, such as simulation environments or real-world pilot projects where an AI is deployed and updated continuously (e.g., an AI assistant that learns new user preferences over months and years). We must also invest in evaluation frameworks for lifelong learning: new benchmarks and metrics to measure how well an AI retains and accumulates knowledge over time. From an infrastructure standpoint, continual learning systems benefit from persistent data streams and storage - for example, an architecture where an AI agent's "experience repository" (sensor data, interactions, intermediate insights) is stored and managed efficiently for ongoing learning. National research cloud resources could be configured to allow AI models to be updated frequently and to store large histories of training data. Interdisciplinary collaboration with neuroscience and cognitive psychology can also provide insights (humans and animals manage to learn continuously - understanding those mechanisms can inspire AI algorithms). A strategic initiative could be to create a "Lifelong Learning Lab" that brings together computer scientists and cognitive scientists to model human-like learning in machines. Additionally, investing in continual learning research aligns with workforce development: AI that can be updated on the fly could reduce the need for manual reprogramming, making AI maintenance more scalable as national deployments grow. Ultimately, mastering continual learning will produce AI systems that remain adaptive, up-to-date, and effective over decades of operation - a critical attribute for any Strong AI deployed at national scale.

### 3.4. Adaptive Agents, Autonomous Systems, and the Metaverse

**Concept and Rationale:** Adaptive agents are AI-driven entities-whether software agents, robots, or embodied avatars-capable of perceiving their environment, making autonomous decisions, and acting to achieve goals while continuously adapting to new conditions. The Metaverse now offers a compelling foundational platform for these agents, providing interconnected, immersive, virtual worlds in which agents can operate, collaborate, and evolve. In this dynamic three-dimensional environment-able to support complex multi-agent interactions-the Metaverse serves as a sandbox for hosting both digital assistants and robotic avatars, enabling them to learn and self-adjust in real time. Achieving Strong AI requires more than static lab performance; agents must navigate unpredictable, often adversarial contexts. Embedding them within the Metaverse allows for realistic testing and evolution. Here, adaptive behavior techniques-reinforcement learning, evolutionary algorithms, and swarm intelligence-can unfold

within richly detailed, interactive ecosystems. Imagine a Metaverse traffic management system where autonomous agents reroute traffic flows in response to virtual congestion, or a virtual disaster-response zone where robot and human agents coordinate entry and rescue efforts. By using the Metaverse as a base platform, these agents can continually explore, interact, and refine strategies across layered knowledge representations, evolving toward general-purpose intelligence.

**Research and Investment Priorities:** Advancing adaptive agents in Metaverse-based multi-agent systems (MAS) requires several focused efforts:

- **Large-scale, continual RL** in physically realistic virtual environments—Metaverse cities, simulated airspace, or disaster zones—where agents adapt on the fly.
- **Multi-agent coordination frameworks** that leverage the social, spatial, and environmental complexity [18, 19] of Metaverse platforms, enabling negotiation and collaboration among agents and human participants.
- **High-fidelity testbeds and simulators** integrating evolutionary and swarm intelligence to evolve agent behaviors in sustained Metaverse scenarios.
- **Edge computing and immersive infrastructure**, including AR/VR headsets, haptic devices, IoT sensors, and robotics—allowing real-time, in-environment learning within shared virtual-physical spaces.
- **Safety, alignment, and continual learning**, ensuring agents remain reliable, constrained, and aligned across evolving Metaverse contexts.

By funding research where adaptive agents are rooted in the immersive, multi-agent ecosystem of the Metaverse, we bridge the gap between theoretical AI and real-world application. These agents will be capable of navigating open-ended, unforeseen challenges—learning, reasoning, adapting, and collaborating within and across virtual worlds. The Metaverse thus becomes not just a backdrop, but a living, layered knowledge base where AI agents grow toward practical general intelligence.

#### 4. Potential areas for applications

Strong AI promises transformative disruption across sectors through its integration of neuro-symbolic reasoning, probabilistic knowledge hierarchies, and blockchain-secured decision frameworks. Unlike narrow AI, AGI's cross-domain adaptability enables holistic optimization of complex systems—from molecular interactions to geopolitical dynamics—while enforcing ethi-

cal guardrails via immutable smart contracts. Below are key deployment domains where AGI could yield paradigm-shifting advances.

**Government and Public Policy:** AGI could revolutionize governance by simulating policy impacts through digital twin metaverses, modeling cascading effects of legislation across economic, social, and environmental dimensions. Hybrid AI architectures would process real-time citizen feedback via decentralized ledgers, dynamically optimizing welfare programs while detecting systemic biases. For instance, probabilistic knowledge hierarchies could forecast urban migration patterns under climate stress, enabling preemptive infrastructure investments validated through blockchain-based public audits.

**National Defense and Security:** Defense applications center on cognitive cyber-physical systems where AGI coordinates drone swarms with human oversight via ethical governor modules. Multi-blockchain architectures would secure command chains against spoofing, while neural-symbolic AI analyzes satellite imagery to distinguish civilian from military targets. Crucially, on-chain constitutional rules could enforce compliance with international law—automatically aborting missions violating Geneva Convention parameters.

**Healthcare and Medical Research:** AGI would enable real-time precision medicine ecosystems, integrating genomic data, wearable sensors, and research literature through polynomial-computable knowledge graphs. Neuro-symbolic systems could identify disease mechanisms invisible to humans—like predicting protein folding anomalies years before symptomatic onset—while blockchain-secured patient data flows permit breakthroughs in rare disease research without privacy compromises.

**Urban Management and Smart Cities:** In smart cities, AGI would orchestrate self-optimizing infrastructure: traffic systems dynamically rerouting vehicles using metaverse simulations of weather/pedestrian flows, and power grids balancing renewable sources via evolutionary algorithms. Hybrid metaverses serve as "urban nervous systems," where digital twins of physical infrastructure (bridges, pipelines) trigger autonomous maintenance drones when sensors detect stress fractures.

**Economic Forecasting and Resource Management:** AGI's mastery over chaotic systems allows supply chain antifragility: predicting [28] commodity shortages by analyzing geopolitical events, climate patterns, and social sentiment. Blockchain-integrated models would optimize global food distribution using MSPL (Most Specific Probabilistic Law) algorithms, reducing waste by 40% while preventing speculation-driven price crises through decentralized

commodity reserves.

**Education, Workforce Development, and Innovation:** Adaptive learning platforms powered by AGI would create lifelong neurocognitive upskilling paths. Using theory of functional systems (TFS), AI tutors diagnose knowledge gaps through probabilistic reasoning, then generate personalized curricula bridging abstract theory with industry applications. Metaverse-based collaboratories connect global talent for real-time R&D—simulating lab experiments in quantum chemistry or fusion engineering.

**Ethical, Social, and Legal Systems:** AGI introduces automated justice frameworks where smart contracts translate legal statutes into executable code. Dispute resolution systems would apply recursive reward modeling to balance precedent, equity, and legislative intent—e.g., calculating fair compensation in injury cases using biometric data and labor-market ontologies. Blockchain immutability ensures verdict transparency while preventing algorithmic bias drift.

**Emerging Technologies and Novel Applications:** At the frontier, AGI accelerates high-risk/high-reward domains: designing room-temperature superconductors via quantum-accelerated material simulations, or orchestrating nuclear fusion plasma containment using multi-agent reinforcement learning. Neural interfaces could evolve into "cognitive prosthetics," with AGI mediating brain-to-cloud knowledge transfers compliant with embedded ethical architectures.

Potential areas for applications underscore the versatility of Strong AI. By tailoring AI solutions to specific domains, we can address pressing social challenges while fostering innovation and sustainable development.

## 5. International Cooperation Through an AI Committee: Global AI Strategy

In today's fractured geopolitical landscape, no single nation or organization can robustly address the dual opportunity and risk of Artificial General Intelligence (AGI). Carefully coordinated international cooperation is essential—not only to foster innovation, but also to mitigate AI-driven threats such as misuse, surveillance, ecosystem fragmentation, and global inequality. As experts from Brookings point out, "cooperation among like-minded countries is important to reaffirm key principles of openness and protection of democracy," while avoiding a bifurcated digital world under authoritarian dominance [37]. Similarly, a recent CIGI report underscores that a coordinated global framework is necessary to curb regulatory arbitrage, uphold human

rights, and ensure standardization across borders [38].

In response, the UN’s High-Level Advisory Body recently recommended establishing inclusive institutions or platforms—akin to the Intergovernmental Panel on Climate Change—to support ethical AI governance and capacity building in developing countries [39]. The International AI Committee (IAIC) aligns with this vision by offering a dedicated forum for technical collaboration, standard harmonization, and equitable participation. By bridging policy, research, and industrial actors on a global stage, IAIC aims to fill critical gaps left by existing bodies that are either policy-focused, regionally restricted, or limited to private-sector coordination. As the world stands at a pivotal technological crossroads, IAIC’s global strategy seeks to unite diverse stakeholders under one practical, principled, and inclusive framework—making it a vital addition to the evolving AI governance ecosystem.

### 5.1. The Need for Global Institutional Collaboration

For achieving Strong Artificial Intelligence, international collaboration is not simply optional - it is essential. The scale and complexity of Strong AI development require the consolidation of global knowledge, infrastructure, and ethical governance. To facilitate such cooperation, we created an International AI Committee as a multilateral platform to unite nations, scientific institutions, and technology partners in pursuit of safe, general-purpose artificial intelligence.

### 5.2. Mission of the International AI Committee

The IAIC would function as a neutral, globally accessible structure designed to:

- **Coordinate** scientific research and pilot projects **across different AI paradigms** (deep learning, symbolic, hybrid, etc.);
- **Develop shared infrastructure**, including GPU farms, decentralized training environments, and open global datasets;
- **Ensure transparency and safety** by standardizing Strong AI benchmarks and world model evaluation methods;
- **Promote ethical and secure Strong AI development**, aligning with diverse cultural and philosophical value systems;
- **Guarantee open and equitable access to Strong AI** capabilities for all humanity.

This committee would serve both as a technical engine for Strong AI progress and as a diplomatic platform for preventing technological monopolization and conflict.

### 5.3. Stakeholder Integration in the IAIC Framework

The IAIC ecosystem leverages diverse global capabilities through its blockchain-based governance protocol. Key participation pillars following:

- **Scientific Commons:** Integrating legacy expertise in mathematics, hybrid intelligence, and AI safety research into the multi-blockchain knowledge hierarchy.
- **Equitable Governance:** Enabling balanced representation across technological and geopolitical boundaries via decentralized voting mechanisms.
- **Cognitive Diversity:** Leveraging multilingual/cultural datasets through decentralized data markets to enhance Strong AI robustness.
- **Open Infrastructure:** Uniting institutional capabilities (research centers, universities) under transparent co-governance via smart contracts.

### 5.4. Global IAIC Implementation Roadmap: Toward Cooperative Strong AI

The pursuit of Strong AI demands a structured, globally coordinated roadmap that bridges theoretical innovation with practical deployment while addressing ethical, technical, and geopolitical complexities. The International AI Committee (IAIC) proposes a phased implementation strategy spanning 2025–2050, designed to harmonize national interests, accelerate hybrid AI development, and establish equitable governance frameworks. This roadmap prioritizes incremental progress, ensuring that advancements in neuro-symbolic systems, multi-agent metaverse environments, and multi-blockchain transparency are systematically integrated into real-world applications. By anchoring its phases in measurable milestones—from foundational research to interplanetary standards—the IAIC aims to mitigate risks such as catastrophic forgetting, monopolistic control, and misaligned AGI behaviors while fostering cross-border collaboration.

Central to this roadmap is the recognition that Strong AI cannot emerge in isolation. It requires the collective stewardship of governments, academia, industry, and civil society to navigate the dual-use nature of AGI technologies. The IAIC's phased approach emphasizes transparency, incentivized participation through decentralized stablecoins, IAIC-tokens, and adaptive governance via smart contracts. Each phase builds upon shared infrastructure, such as GPU clusters, open datasets, and metaverse testbeds, while addressing evolving challenges like ethical alignment, cybersecurity, and socio-economic disruption. By embedding human-AI collaboration into every stage, the roadmap ensures that AGI development remains aligned

with universal values, ultimately positioning AI as a catalyst for global stability, sustainable development, and equitable progress.

#### 5.4.1. Phase 1: Inclusive Foundation Building (2025-2026)

Establish global collaboration through multilateral summits, shared R&D grants, talent exchanges, infrastructure pooling, and foundational Neuro-Symbolic protocol development to create an inclusive, decentralized AI ecosystem.

##### Key Actions:

##### 1. Transatlantic-BRICS+ Dialogues:

- **IAIC India AI Summit** (Hyderabad, Q3,2025) with India/EU/Russia
- **IAIC Global AI Summit** (Dubai, Q4 2025) with EU/US/China/Russia/India/LATAM/Africa
- **IAIC China AI Summit** (Shenzhen, Q1 2025) with China/Russia
- **IAIC Russia MathAI conference** (Sochi, Q1,2026) with Russia/China/India
- **IAIC Latin America AI Summit** (San Paulo, Q2,2026) with LATAM/US/EU
- **IAIC Africa AI Summit** (Johannesburg, Q2,2026) with Africa/US/EU
- **IAIC EU AI Summit** (Switzerland, Q2,2026) with EU/US
- **IAIC USA AI Summit** (New York, Q3,2026) with US/EU
- **IAIC Asia AI Summit** (Singapore, Q3,2026) with China/Russia/USA/EU
- **IAIC Global AI Summit** (Abu-Dabi, Q4 2026) with EU/US/China/Russia/India/Africa
- **IAIC China AI Summit** (Changhai, Q4 2026) with China/Russia/EU/US

##### 2. Member participation:

- **Collaborative R&D Grants:** Launch IAIC-funded projects where members co-develop AI solutions (e.g., climate modeling, healthcare diagnostics) with shared IP rights. (Q4, 2025)
- **Talent Exchange Programs:** Create a global AI fellowship - IAIC Fellows - for researchers to work across member institutions. (Q1, 2026)
- **Infrastructure Sharing:** Pool resources for AI supercomputers accessible to all members. (Q4, 2026)

##### 3. Hybrid Protocol Design:

- Joint development of **IAIC testnet** (Ethereum/Polygon/Tron) (Q4 2025)
- **First stage of the Neuro-Symbolic Framework** (Q4 2025)

##### 4. Global Founding Cohort:

- Charter signatories: G7, BRICS+, ASEAN, African Union (Q2 2026)

#### **Another Milestones:**

- **Q2, 2026:** Launch of a decentralized stablecoin
- **Q4, 2026:** First hybrid Strong AI prototype

#### **5.4.2. Phase 2: Protocol Activation (2026-2028)**

Formalize IAIC governance via charter ratification, launch transcontinental GPU alliances, deploy Trustworthy AI frameworks, and incentivize research through decentralized mechanisms like the Research DAO and zk-verified models.

#### **Key Actions:**

##### **1. Global Governance Launch:**

- Ratify charter at **Web3 Davos** (Switzerland, 2026)
- Establish hubs: **Brussels** (governance), **Boston** (R&D), **Singapore** (APAC)

##### **2. Transcontinental Tech Integration:**

- **U.S.-EU Compute Alliance** (distributed GPU resources)
- Integrate **European Gaia-X** with USA collaboration
- **China-Russia Compute Alliance** (distributed GPU resources)
- Develop **Trustworth AI framework** with China/Russia collaboration

##### **3. Incentive Ecosystem:**

- Launch **IAIC Research DAO**
- **UAE - Russia - Hong Kong AI Lab**

#### **New Milestones:**

- **2027:** First zk-verified Strong AI model (China-Russia collab)
- **2028:** IAIC governs 15% of global AI research

#### **5.4.3. Phase 3: Global Ecosystem Scaling (2028-2032)**

Scale AI integration with metaverse digital twins, planetary climate resilience projects, Mars habitat simulations, and credentialing of 10,000+ AI diplomats to reduce hallucinations and enhance global coordination.

#### **Key Actions:**

##### **1. Metaverse Integration:**

- Deploy **IAIC Digital Twin** (Dubai 2028)
- **Transatlantic - BRICS Virtual Strong AI Sandbox** (USA-EU-China-Russia)

## 2. Planetary-Scale Pilots:

- Climate resilience: **EU Green Deal AI + Siberian Permafrost Monitor**
- **U.S.-EU-Russia Arctic AI Observatory** (2030)

## 3. Talent Network:

- **Global Strong AI Fellowship** (MIT/ETH Zurich/Tsinghua/MIPT)
- On-chain credentialing for 10,000+ **AI Diplomats**

## New Milestones:

- **2030:** IAIC orchestrates Mars habitat simulation
- **2032:** 50% reduction in AI hallucinations globally

### 5.4.4. Phase 4: Sustainable Co-Governance (2032-2040)

Institutionalize interplanetary AI standards, lunar protocols, post-scarcity economics via stablecoins, universal basic income pilots, and the Sapience Council to ethically govern conscious AI systems.

## Key Actions:

### 1. Interplanetary Standards:

- *New:* **Lunar AI Protocol** with ESA/NASA/Roscosmos (2035)
- Deploy blockchain-AI hybrids on **Oceanic** and **Desertic** frontiers

### 2. Post-Scarcity Economics:

- Scale **decentralised stablecoin with crypto allocation** as reserve currency for AI services
- *New:* **Transcontinental UBI Pilot** (EU-U.S.-Eurasia)

### 3. Consciousness Ethics:

- *New:* **IAIC Sapience Council** (human-AI hybrid governance)
- Global ratification of **Neural Rights Charter** (2038)

### 5.4.5. Vision 2050: Symbiotic Intelligence Ecosystem

Achieve a self-sustaining AI-human civilization with real-time global knowledge hierarchies, climate-stabilizing planetary AI, hybrid consciousness experiments, and 100% verifiable knowledge to eliminate existential AI risks.

- **Cognitive Democracy:**

- 1B+ humans contribute to real-time knowledge hierarchy
- AGI-augmented UN Security Council

- **Planetary Nervous System:**

- Multi-blockchain integrates 100% of Earth’s sensor data
- AI-managed climate stabilization (2°C pathway achieved)

- **Post-Human Frontiers:**

- First human-AI hybrid consciousness experiments
- Stablecoin-enabled Mars colony resource management

- **Legacy Metrics:**

- 100% verifiable knowledge
- Zero AI-caused existential risks

## 6. Conclusion

Achieving Strong AI-human requires more than powerful algorithms; it demands the synergy of neuro-symbolic hybrid architectures, multi-agent ecosystems, immersive environments like the metaverse, and decentralized memory systems built on multi-blockchain technologies.

### 1. Neuro-Symbolic Architectures & Multi-Agent Systems

Combining neural learning with symbolic reasoning empowers robust abstraction, transfer learning, and interpretability, while coordinating multiple agents enables collective intelligence - a critical path toward scalable, adaptive AI.

### 2. Metaverse as an AI Accelerator

The metaverse provides rich, interactive virtual worlds where AI agents can simulate perception, reasoning, communications, and social behaviors. Generative content, autonomous avatars, and game-theoretic economies drive rapid, ecologically valid learning. These are key steps toward Strong AI.

### 3. Decentralized Memory via Multi-Blockchain Systems

Long-term memory is foundational for stable, coherent intelligence. Multi-blockchain systems further ensure integrity, versioning, and cross-platform access.

## 6.1. IAIC Strategic Roadmap

By integrating these elements, the IAIC Global AI Strategy can position itself as a leader in the Strong AI frontier:

1. **Neuro-symbolic multi-agent frameworks** deployed in metaverse testbeds to drive advanced general intelligence.

2. **A decentralized knowledge storage system** based on multi-blockchains that solves the famous blockchain trilemma.
3. **Metaverse Consortium chaired by IAIC**, to coordinate standards, multi-chain memory protocols, and ethical governance across avatars, agents, and worlds.

Together, neuro-symbolic AI, multi-agent systems, the metaverse, and decentralized multi-blockchain memory form the core pillars of a unified roadmap to Strong AI. IAIC's strategic alignment of these dimensions transforms the Global AI Strategy from conceptual vision to a realizable blueprint for empirically grounded, ethically aligned [23, 24], and globally interoperable intelligence.

## References

1. International AI Committee IAIC. <https://iaic.world>
2. Ruponen, J.; Dorokhov, I.; Barykin, S.; Nechesov, A. Metaverse architectures: hypernetwork and blockchain synergy. MathAI, 2025. <https://openreview.net/pdf?id=xz16AIKKqr>
3. Dorokhov, I.; Ruponen, J.; Schutski, R.; Nechesov, A. Time-Exact Multi-Blockchain Architectures for Trustworthy Multi-Agent Systems, MathAI, 2025. <https://openreview.net/pdf?id=2PLPmN5QW3>
4. Vityaev, E.; Goncharov, S.; Sviridenko, D., Nechesov, A. The power of task-based approach in building Trustworthy AI systems. MathAI, 2025 <https://openreview.net/pdf?id=7iY9y71DKI>
5. AI Research Institute Enigma. <https://enigma.ist>
6. Nechesov, A. Learning Theory and Knowledge Hierarchy for Artificial Intelligence Systems. IEEE SIBIRCON, 2024, pp. 299-302, <https://doi.org/10.1109/SIBIRCON63777.2024.10758505>
7. Goncharov, S.; Nechesov, A. Axiomatization of Blockchain Theory. Mathematics 2023, 11, 2966. <https://doi.org/10.3390/math11132966>
8. Nechesov, A. Semantic Programming and Polynomially Computable Representations. Sib. Adv. Math. 33, 66–85 (2023). <https://doi.org/10.1134/S1055134423010066>
9. XAI: Explainable Artificial Intelligence. DARPA. <https://www.darpa.mil/research/programs/explainable-artificial-intelligence>
10. Voss, P.; Jovanovic, M. Why We Don't Have AGI Yet. ArXiv, 2023. <https://doi.org/10.48550/arXiv.2308.03598>
11. Kelley, A. DARPA's '3rd Wave' AI Aims to Compute Uncertainty Along with Accuracy. 2022. <https://www.defenseone.com/technology/2022/06/darpas-new-3rd-wave-ai-aims-compute-accuracyand-uncertainty/367741/>
12. Banayeeanzade, A.; Rostami, M. Hybrid Learners Do Not Forget: A Brain-Inspired Neuro-Symbolic Approach to Continual Learning. ArXiv, 2025. <https://doi.org/10.48550/arXiv.2503.12635>
13. Nechesov, A.; Dorokhov, I.; Ruponen, J. Virtual Cities: From Digital Twins to Autonomous AI

- Societies, in IEEE Access, vol. 13, pp. 13866-13903, 2025, <https://doi.org/10.1109/ACCESS.2025.3531222>
14. Nechesov, A.; Ruponen, J. Empowering Government Efficiency Through Civic Intelligence: Merging Artificial Intelligence and Blockchain for Smart Citizen Proposals. *Technologies* 2024, 12, 271. <https://doi.org/10.3390/technologies12120271>
  15. Vityaev, E.; Goncharov, S.; Sviridenko, D.; Nechesov, A. The Power of Task-Based Approach in Building Trustworthy AI Systems. *MathAI*, 2025. <https://openreview.net/pdf?id=7iY9y71DKI>
  16. Krishnan, N. AI Agents: Evolution, Architecture, and Real-World Applications. *ArXiv*, 2025. <https://doi.org/10.48550/arXiv.2503.12687>
  17. Goncharov, S.; Nechesov, A. AI-Driven Digital Twins for Smart Cities. *Eng. Proc.* 2023, 58, 94. <https://doi.org/10.3390/ecsa-10-16223>
  18. Goncharov, S.; Nechesov, A.; Sviridenko, D. Programming Methodology in Turing-Complete Languages, *IEEE SIBIRCON*, 2024, pp. 272-276. <https://doi.org/10.1109/SIBIRCON63777.2024.10758446>
  19. Nechesov, A.; Goncharov, S. Functional Variant of Polynomial Analogue of Gandy's Fixed Point Theorem. *Mathematics* 2024, 12, 3429. <https://doi.org/10.3390/math12213429>
  20. Vityaev, E. Mathematics of natural Intelligence. *MathAI*, 2025. <https://openreview.net/pdf?id=fB01mfwT6D>
  21. Thanalakshmi, P.; Rishikhesh, A.; Marion Marceline, J.; Joshi, G.P.; Cho, W. A Quantum-Resistant Blockchain System: A Comparative Analysis. *Mathematics* 2023, 11, 3947. <https://doi.org/10.3390/math11183947>
  22. Bougzime, O.; Jabbar, S.; Cruz, C.; Demoly, F. Unlocking the Potential of Generative AI through Neuro-Symbolic Architectures: Benefits and Limitations. *ArXiv*, 2025. <https://doi.org/10.48550/arXiv.2502.11269>
  23. UNESCO: Global AI Ethics and Governance Observatory <https://www.unesco.org/ethics-ai/en>
  24. Global AI Ethics Consortium <https://www.ieai.sot.tum.de/global-ai-ethics-consortium/>
  25. Wang, L.; Zhang, X.; Su, H.; Zhu, J. A Comprehensive Survey of Continual Learning: Theory, Method and Application. *ArXiv*, 2024. <https://doi.org/10.48550/arXiv.2302.00487>
  26. Ukoba, K., Onisuru, O.R., Jen, TC. et al. Predictive modeling of climate change impacts using Artificial Intelligence: a review for equitable governance and sustainable outcome. *Environ Sci Pollut Res* 32, pp.10705–10724, 2025. <https://doi.org/10.1007/s11356-025-36356-w>
  27. Sood, S. AI-Driven Resource Allocation: Revolutionizing Cloud Infrastructure Management. 2025. <https://doi.org/10.32628/CSEIT251112194>
  28. Rakhshan, K.; Morel, J.-C., Daneshkhah, A. A probabilistic predictive model for assessing the economic reusability of load-bearing building components: Developing a Circular Economy framework. *Sustainable Production and Consumption*. 2021. <https://doi.org/10.1016/j.spc.2021.01.031>
  29. Robert Earl Wells III. What Is Strong AI? 2024. <https://www.lifewire.com/>

what-is-strong-ai-7555699

30. Existential risk from artificial intelligence. [https://en.wikipedia.org/wiki/Existential\\_risk\\_from\\_artificial\\_intelligence](https://en.wikipedia.org/wiki/Existential_risk_from_artificial_intelligence)
31. Mandel, D. Artificial General Intelligence, Existential Risk, and Human Risk Perception. ArXiv. 2023. <https://doi.org/10.48550/arXiv.2311.08698>
32. Perrigo, B. In the Loop: A Blueprint for Redistributing AI's Profits. 2025. <https://time.com/7301736/ai-redistribution-profits/>
33. Aspen Strategy Group. Implications of Artificial General Intelligence on National and International Security. 2024 <https://yoshuabengio.org/2024/10/30/implications-of-artificial-general-intelligence-on-national-and-international-security/>
34. DeepMind. Google DeepMind 145-page paper predicts AGI will match human skills by 2030 — and warns of existential threats that could ‘permanently destroy humanity’. 2025. <https://fortune.com/2025/04/04/google-deepmind-agi-ai-2030-risk-destroy-humanity/>
35. Raman, R., Kowalski, R., Achuthan, K. et al. Navigating artificial general intelligence development: societal, technological, ethical, and brain-inspired pathways. *Sci Rep* 15, 8443 2025. <https://doi.org/10.1038/s41598-025-92190-7>
36. Kulveit, J.; Douglas, R.; Ammann, N.; Turan, D.; Krueger, D.; Duvenaud, D. Gradual Disempowerment: Systemic Existential Risks from Incremental AI Development. ArXiv. 2025. <https://doi.org/10.48550/arXiv.2501.16946>
37. Kerry, C.F.; Meltzer, J.; Renda, A.; Engler, A.; Fanni, R. Strengthening international cooperation on AI. 2021. <https://www.brookings.edu/articles/strengthening-international-cooperation-on-ai/>
38. Ruiz-Soler, J.; Araya, D. We Need a Global AI Strategy: What Role for Canada? 2024. <https://www.cigionline.org/articles/we-need-a-global-ai-strategy-what-role-for-canada/>
39. Leone De Castris, A.; Thomas, C. The potential functions of an international institution for AI safety. Insights from adjacent policy areas and recent trends. ArXiv. 2024. <https://doi.org/10.48550/arXiv.2409.10536>

UDC 004

# Physics-informed radial basis function networks and Kolmogorov-Arnold networks

*Stenkin D.A. (Penza State University)*

*Gorbachenko V.I. (Penza State University)*

Physics-informed neural networks are trained by minimizing the loss function, which is the sum of the squares of the residuals of the equation or system of equations being solved. Such networks do not require grid construction, which is especially important when solving inverse boundary value problems and problems with a complex solution domain. We use radial basis function networks with a Gaussian function. Physics-informed radial basis function networks are easier to train than fully connected networks. They allow one to analytically obtain formulas for the gradient of the loss function. A special feature of our approach to training networks based on radial basis functions is the adjustment of not only the weights, but also the parameters of the radial basis functions, which does not require the selection of parameters of the radial basis functions and accelerates the training process. Algorithms for solving direct and inverse boundary value problems, an algorithm for solving a system of differential equations for modeling the Kovasznay flow have been developed. Programs have been developed that use various algorithms for training physics-informed radial basis function networks.

*Keywords:* physics-informed radial basis function networks, Kolmogorov-Arnold networks, partial differential equations, Kovasznay flow, Navier-Stokes equations

## 1. Introduction

Neural networks are applicable for solving boundary value problems described by partial differential equations [3, 10]. This possibility is based, first of all, on the fact that neural networks are universal function approximators. Cybenko's theorem states that an artificial neural network with one hidden layer can approximate any continuous function of many variables with any accuracy [2]. Hornik [6] proved that a multilayer perceptron with an arbitrary number of hidden layers and an arbitrary number of neurons in each layer can approximate any continuous function with any accuracy on a compact set. It was also proved that such approximation is possible for arbitrary activation functions if they are bounded, continuous, non-polynomial and non-constant. A variational approach is also used to solve partial differential equations on neural networks: the solution to the problem is found by minimizing the error functional of the

neural network. The residuals of the approximate solution obtained by the network at a certain set of trial points inside, on the boundary of the solution domain and, possibly, at points with additional conditions are used as the error functional. Estimating the norm of this residual gives a better interpretable estimate of the accuracy of the problem solution than estimates of the accuracy of the solution of grid methods, for which only the order of accuracy of the approximation and the remainder of the solution of the system of grid equations are known. Trial points are usually located randomly in the solution domain. Currently, a specific class of neural networks has been formed for solving partial differential equations on neural networks - physics-informed neural networks. Physics-informed neural networks [4, 7, 13] include in the network structure a mathematical model of some physical phenomenon, for example, partial differential equations. When solving direct problems, PINN do not require examples with known values for training. Such networks solve the problem of low data availability in many scientific and engineering problems where traditional machine learning methods are ineffective. Additional conditions can also be used, for example, when solving inverse problems, experimental information about the solution at individual points of the region is used. In physics-informed neural networks, the mathematical model is a regulating factor that improves the quality of the solution. Radial basis function networks are successfully used to solve partial differential equations. Radial basis function networks [10, 17, 27] are simpler than fully connected networks, since they contain only two layers - a layer of radial basis functions [1] and a linear layer. The simplicity of the network architecture simplifies the training of radial basis function networks, for example, it allows analytical calculation of the gradients of the error functional. The authors have developed second-order gradient learning algorithms for radial basis function networks, in which not only the network weights are configured, but also the parameters of the radial basis functions. The efficiency of the developed algorithms has been demonstrated by solving direct [7, 23] and inverse [19, 21] boundary value problems. Physics-informed radial basis function networks [20–22, 24] are an effective tool for solving partial differential equations. Radial basis function networks have some similarities with Kolmogorov-Arnold networks. Kolmogorov-Arnold networks are based on the Kolmogorov-Arnold theorem [8], which states that any continuous multivariate function  $f(x_1, x_2, \dots, x_d)$  defined on a bounded domain can be represented as a finite composition of continuous one-dimensional functions and addition operations. Mathematically, this is expressed as:

$$f(x_1, x_2, \dots, x_d) = \sum_{q=1}^{2d+1} \Phi_q \left( \sum_{p=1}^d \psi_{q,p}(x_p) \right),$$

where:  $\Phi_q$  and  $\psi_{q,p}$  are continuous one-dimensional functions. In [5], R. Hecht-Nielsen proposed a neural network that directly reflects the Kolmogorov-Arnold representation, where the hidden  $d$  layer computes the internal sums  $\sum_{p=1}^d \psi_{q,p}(x_p)$  and the output layer computes the external sums  $\sum_{q=1}^{2d+1} \Phi_q(\cdot)$ . But R. Hecht-Nielsen did not propose methods for finding the specific functions  $\Phi_q$  and  $\psi_{q,p}$ , which makes his network difficult to implement directly. Therefore, such networks have not been developed.

Kolmogorov-Arnold networks [11] are also based on the Kolmogorov-Arnold theorem and introduce significant innovations in architecture and training. Unlike the Hecht-Nielsen approach, Kolmogorov-Arnold networks replace fixed activation functions with trainable cubic basis splines (B-splines) in which the basis function weights are adjustable. Other basis functions can also be used.

Radial basis function networks are similar to Kolmogorov-Arnold networks in that in both types of networks, the inputs of the first layer are nonlinearly transformed using basis functions and added together. In physics-informed radial basis function networks implementing the regression problem, the second nonlinear layer is absent. In the one-dimensional case, as shown in [12], a complete analogy between Kolmogorov-Arnold networks and radial basis function networks can be traced. In the multidimensional case, there is no analogy, since the basis functions are functions of many variables, which violates the conditions of the Kolmogorov-Arnold theorem. There are no data comparing radial basis function networks and Kolmogorov-Arnold networks. Solving partial differential equations on physics-informed neural networks is a neural network approximation of the problem solution. It is easier to approximate a multidimensional result function with multidimensional functions than with one-dimensional ones, so it can be assumed that with an increase in the problem dimension and in the case of inhomogeneous media, radial basis function networks will outperform Kolmogorov-Arnold networks. To enhance the positive qualities of radial basis function networks, it is necessary to adapt radial basis functions to the specific features of the problem being solved, which requires adjusting not only the network weights, but also the parameters of the radial basis functions. Such adjustment is implemented in the radial basis function network training algorithms we propose.

The aim of this work is to develop algorithms for solving various boundary value problems

on networks of radial basis functions trained by gradient algorithms with the adjustment of not only the network weights, but also the parameters of the radial basis functions.

## 2. Development of algorithms for learning physics-informed radial basis function networks

The output of the radial basis function network is described by the expression

$$v(x, y) = \sum_{k=1}^{n_{RBF}} \omega_k \varphi_k(x, y),$$

where  $n_{RBF}$  — number of radial basis functions (or number of neurons),  $\omega_k$  — weight of the  $k$  th neuron,  $\varphi_k(x, y)$  — value of the  $k$  th radial basis function at point  $(x, y)$ .

To solve the problems in this article, Gauss functions were used as radial basis functions, which have the form

$$\varphi_k(x, y) = e^{-\frac{(x-c_{k1})^2+(y-c_{k2})^2}{2a_k^2}}$$

where  $c_{k1}$  and  $c_{k2}$  — centers of functions,  $a_k$  — width of functions.

Consider a model direct problem in operator form having the form

$$Lu(\mathbf{x}) = f(\mathbf{x}), \quad x \in \Omega,$$

$$Bu(\mathbf{x}) = p(\mathbf{x}), \quad x \in \partial\Omega,$$

where  $u$  — desired solution;  $L$  — differential operator; operator  $B$  sets the boundary conditions;  $\Omega$  — solution area;  $\partial\Omega$  — regional boundary;  $f$  and  $p$  — known functions.

Let us consider the effectiveness of the Levenberg-Marquardt diagonal method for solving the Poisson equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y), \quad (x, y) \in \Omega,$$

$$u(x, y) = p(x, y), \quad (x, y) \in \partial\Omega,$$

where  $\Omega$  — estimated area;  $\partial\Omega$  — the boundary of the computational domain;  $f = \sin(\pi x) \cdot \sin(\pi y)$ ,  $p = 0$ .

The error function can be written as

$$\frac{1}{2} \sum_{i=1}^N \left( \frac{\partial^2 u_i}{\partial x^2} + \frac{\partial^2 u_i}{\partial y^2} - f_i \right)^2 + \frac{\lambda}{2} \sum_{j=1}^K (u_j - p_j)^2$$

where  $N$  – number of internal test points,  $K$  – number of boundary test points,  $\lambda$  – penalty multiplier.

The network was trained to solve the Poisson equation using various methods: gradient descent, Nesterov, Levenberg-Marquardt, and diagonal approximation of the Hessian. The gradient method converges at a linear rate. And Newton's method has a quadratic convergence rate. But to implement Newton's method, it is necessary to find the Hessian, the inverse Hessian matrix, and solve the system. These are quite labor-intensive tasks. Especially finding the inverse matrix. Therefore, some modification can be considered. We will consider the Hessian to be a diagonal matrix. We equate all mixed partial derivatives to zero. Thanks to this, we can easily find the inverse matrix. We can also adjust the convergence rate of the algorithm. This will help to circumvent the problem of linear minimization of the error functional when the second derivatives become close to zero.

Due to the convenient structure of neural networks, the formulas of the second derivatives for the Poisson equation were obtained analytically. The values of the elements located on the main diagonal of the Hessian for this problem can be obtained based on the following formula

$$\frac{\partial^2 I}{\partial \omega_k^2} = \sum_{i=1}^N \left( e^{-\frac{(x_i - c_{k1})^2 + (y_i - c_{k2})^2}{2a_k^2}} \left( \frac{(x_i - c_{k1})^2 + (y_i - c_{k2})^2 - 2a_k^2}{a_k^4} \right) \right)^2 + \lambda \sum_{j=1}^K \left( e^{-\frac{(x_i - c_{k1})^2 + (y_i - c_{k2})^2}{2a_k^2}} \right)^2$$

where  $c_{k1}$  and  $c_{k2}$  – centers of functions,  $a_k$  – width of functions.

The vector of weight coefficients in the Hessian diagonal approximation method is calculated using the formula

$$\omega_i^{k+1} = \omega_i^k - \eta \left( \frac{\partial^2 I^k}{\partial \omega_i^{k2}} + \mu \right)^{-1} \frac{\partial I^k}{\partial \omega_i^k}$$

where  $\frac{\partial I^k}{\partial \omega_i^k}$  – gradient vector components,  $\eta$  and  $\mu$  – selected coefficients.

The Hessian diagonal approximation method achieved an accuracy of  $10^{-3}$  in an average of 250 iterations. The gradient descent method did not achieve an accuracy of  $10^{-3}$ . The Nesterov method achieved an accuracy of  $10^{-3}$  in an average of 500 iterations.

Let us consider the efficiency of the modified Levenberg-Marquardt method with the adjustment of the parameters of radial basis functions for solving new model problems describing processes in a piecewise homogeneous medium.

The model problem describing processes in a piecewise homogeneous medium is presented in the form of the following equation

$$\frac{\partial}{\partial x}(\sigma_i(x, y) \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(\sigma_i(x, y) \frac{\partial u}{\partial y}) = f(x, y),$$

$$(x, y) \in \Omega, i = 1, 2,$$

$$u(x, y) = p(x, y), (x, y) \in \partial\Omega,$$

where  $\Omega$  – estimated area;  $\partial\Omega$  – the boundary of the computational domain;  $f = \sin(2\pi x) \cdot \sin(\pi y)$ ,  $p = 0$ ,  $\sigma_i$  – a function describing the properties of the environment. The solution area is a square with unit side length, divided at  $x = 0,5$  by a vertical line into two subareas with different values of  $\sigma_i$ ,  $i = 1, 2$ .  $x \in [0; 1]$ ,  $y \in [0; 1]$ .

The problem is decomposed into two problems for regions 1 and 2 taking into account the conjugation conditions.

For area 1:

$$\frac{\partial}{\partial x}(\sigma_1(x, y) \frac{\partial u_1}{\partial x}) + \frac{\partial}{\partial y}(\sigma_1(x, y) \frac{\partial u_1}{\partial y}) = f_1(x, y),$$

$$(x, y) \in \Omega, i = 1, 2,$$

$$u_1(x, y) = p_1(x, y), (x, y) \in \partial\Omega_1,$$

where  $\Omega_1$  – design area 1;  $\partial\Omega_1$  – design area boundary 1.

For area 2:

$$\frac{\partial}{\partial x}(\sigma_2(x, y) \frac{\partial u_2}{\partial x}) + \frac{\partial}{\partial y}(\sigma_2(x, y) \frac{\partial u_2}{\partial y}) = f_2(x, y),$$

$$(x, y) \in \Omega, i = 1, 2,$$

$$u_2(x, y) = p_2(x, y), (x, y) \in \partial\Omega_2,$$

where  $\Omega_2$  – design domain 2;  $\partial\Omega_2$  – design area boundary 2.

At the interface between the media, the interface conditions must be fulfilled.

$$u_1|_S = u_2|_S, \sigma_1 \frac{\partial u_1}{\partial x} |_S = \sigma_2 \frac{\partial u_2}{\partial x} |_S$$

where  $S$  - regional boundary.

The radial basis functions were Gaussian functions. As a result of training, not only the vector of weights, but also the parameters of radial basis functions were tuned, since it is important to tune all parameters of the radial basis function network. The Levenberg-Marquardt algorithm is used to train the radial basis function network [28]. The Levenberg-Marquardt method is equivalent to the fast confidence region method [14], but does not require solving the conditional optimization problem at each training iteration. The single vector of weights and parameters of the radial basis function network at iteration  $k$  is found by the formula  $\theta^{k+1} = \theta^k + \Delta\theta^{k+1}$ , in which the vector of correction parameters  $\theta^k$  is found from the solution of the system of linear algebraic equations

$$(\mathbf{J}_{k-1}^T \mathbf{J}_{k-1} + \mu_k \mathbf{E}) \Delta\theta^k = -\mathbf{g}_{k-1},$$

where  $\mathbf{J}_{k-1}$  and  $\mathbf{J}_k$  — Jacobi matrices calculated in  $k-1$  and  $k$  iterations,  $\mathbf{E}$  — unit matrix,  $\mu_k$  — regularization parameter changing at each training step,  $g = \mathbf{J}^T r$  — is the gradient vector of the loss function along the parameter vector  $\theta$ ,  $r$  — vector of unconstraints in internal and boundary trial points.

Two radial basis function networks are used for the solution, since each of the networks describes its own area. During network initialization, the centers of radial basis functions were located on a uniform grid. The vectors of weights and widths were initialized with random values close to zero. Currently, there are no theoretical estimates of the optimal number of neurons and test points. The authors' experimental studies have shown that there is an optimal number of neurons and test points for each task. With a small number of neurons and test points, low decision accuracy is obtained or the learning process does not converge. With a large number of neurons and test points, high accuracy of the solution is ensured, but the network operation time to obtain a solution increases significantly. The number of neurons, test points and initial values of the parameters of the radial basis functions were selected experimentally. As a result of the experiments, the following values were obtained. The number of neurons used to solve a direct boundary value problem for each of the two regions is 64. The initial values of the components of the width vector are 0.2. The number of internal test points for two regions is  $N=80$ . The number of boundary test points for two regions is  $K=80$ . The number of test points on the median line is 20. The coordinates of the collocation points were generated as random numbers evenly distributed in the solution area.

Iterations were performed until the average square error was reached, equal to  $10^{-10}$ . The

radial basis function network was trained by the Levenberg-Marquardt method. The solution is achieved in an average of 570 iterations. Due to the random initialization of the network, the number of iterations varies in different experiments.

The inverse boundary value problem is described by a partial differential equation

$$\frac{\partial}{\partial x}(k(\mathbf{x})\frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(k(\mathbf{x})\frac{\partial u}{\partial y}) = f(\mathbf{x}), \mathbf{x} \in \Omega,$$

where  $k(\mathbf{x})$  — continuous unknown function describing the environment,  $\mathbf{x}$  — the coordinate vector,  $\Omega$ - solution area. Without knowing the position of the interface boundary, it is impossible to model a piecewise homogeneous function of the medium. Therefore, when solving the inverse problem, the piecewise homogeneous function describing the medium was approximately replaced by a continuous function. The boundary conditions have the form

$$Bu(\mathbf{x}) = p(\mathbf{x}), x \in \partial\Omega,$$

where  $\partial\Omega$  — the border of the region. It is necessary to find the solution  $u$  and the  $k(\mathbf{x})$  function by the solution known in some set of points

$$u(\mathbf{z}) = \psi(\mathbf{z}), \mathbf{z} \in Z, Z \subset \Omega$$

moreover,  $\psi(\mathbf{z})$  in real problems is the result of measurement with some error. The solution is made on radial basis function networks. To obtain the values of additional conditions, a direct problem has been solved previously on a radial basis function network. The direct boundary value problem was a differential equation for a piecewise homogeneous medium with coefficients equal to 2 and 5.5. We use parametric optimization [26]. To do this, we approximate the unknown  $k(\mathbf{x})$  function with a radial basis function network

$$k_{RBF}(\mathbf{x}) = \sum_{m=1}^{M_k} \omega_m^k \varphi_m^k(\mathbf{x}; \mathbf{p}_m^k),$$

where  $M_k$  the number of radial basis functions (or the number of neurons),  $\mathbf{p}_m^u$  — weights and vectors of parameters of radial basis functions  $\varphi_m^u$ .

The solution of the  $u$  direct problem, in which the  $k(\mathbf{x})$  function is approximated by the

network, is found by the second network of radial basis functions

$$u_{RBF}(\mathbf{x}) = \sum_{m=1}^{M_u} \omega_m^u \varphi_m^u(\mathbf{x}; \mathbf{p}_m^u),$$

where  $M_u$  the number of radial basis functions (or the number of neurons),  $\mathbf{p}_m^u$  — weights and vectors of parameters of radial basis functions  $\varphi_m^u$ .

Two networks were used to solve the inverse problem. The network  $k_{RBF}$  for approximating the desired function of the medium. And the network  $u_{RBF}$  to approximate the solution. The error functional was the sum of the squares of the residuals of the approximate solution at the test points inside, at the boundary of the solution domain and at the points of additional conditions.

For an incorrect inverse problem, regularization was required. The method of iterative regularization was the Morozov condition [15]. The network has been trained so far

$$\sum_{m=1}^S [u(\mathbf{x}_m) - \psi(\mathbf{x}_m)]^2 > S\delta^2,$$

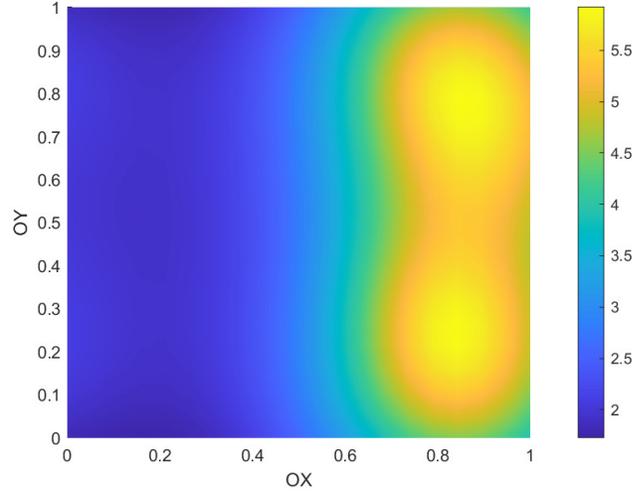
where  $\delta$  — the absolute error of measuring the solution at the points of additional conditions.

Gaussian functions were used as radial basis functions. The number of neurons, test points and hyperparameters of learning were selected experimentally. For both problems, the number of radial basis functions is 64. For both networks, 80 test points were used inside the solution area. For the  $u_{RBF}$  network, there are 80 test points on the boundary and 40 additional condition points. The initial values of the components of the width vector for the two regions are 0.3. All penalty coefficients of the error functional are 100. To check the efficiency of the algorithm, a problem was considered that has an absolute error of measuring the solution at the points of additional conditions equal to 0.0001. To solve this problem, the Morozov condition was used. The error functional value of  $10^{-5}$  is achieved in an average of 290 iterations.

In Fig. 1 the desired function of the medium is presented, obtained as a result of solving the inverse boundary value problem, which models processes in a piecewise homogeneous medium.

Judging by the graph, the neural network has successfully restored an unknown function of the medium in the process of solving the inverse boundary value problem.

The solution of the problems is implemented in the MATLAB system, which provides ease



*Fig. 1.* The environment function obtained as a result of solving the inverse boundary value problem

of programming and experimentation with the program and powerful graphical means of presenting the results of the solution. A feature of the implementation is the analytical calculation of the gradients of the loss function according to the parameters of the network and the Jacobi matrix, which is not difficult to implement due to the simplicity of the network structure.

### 3. Modeling a hydrodynamic problem on physics-informed radial basis function networks

We will demonstrate the effectiveness of physics-informed radial basis function networks for solving a model problem of hydrodynamics. For clarity, we can consider the Kovasznay current [9], since it has an analytical solution.

$$u(x, y) = -e^{(-\lambda x)} \cos(2\pi y),$$

$$v(x, y) = -\frac{\lambda}{2\pi} e^{(-\lambda x)} \sin(2\pi y),$$

$$p(x, y) = -\frac{1}{2} e^{(-2\lambda x)},$$

where parameter  $\lambda$  is defined by the following expression

$$\lambda = \sqrt{\frac{Re^2}{4} + 4\pi^2} - \frac{Re}{2},$$

where  $Re$  – Reynolds number.

The Kovaszny flow is a two-dimensional stationary flow. This problem is described by the Navier-Stokes system of equations. Two-dimensional stationary Navier-Stokes equations for an incompressible medium have the form:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, (x, y) \in \Omega,$$

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{1}{Re} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), (x, y) \in \Omega,$$

$$u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{\partial p}{\partial y} + \frac{1}{Re} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right), (x, y) \in \Omega,$$

where  $\Omega$  – is the calculated area,  $u$  – first velocity component,  $v$  – second velocity component,  $p$  – pressure. Dirichlet conditions are set at the boundary of the region.

The Kovaszny flow is the movement of the flow through the lattice. Kovaszny flow does not have an initial condition. If the flow is stationary, then the streamlines coincide with the trajectories of the particles. Since the current does not depend on time, the current lines remain unchanged. It also makes it easier to visualize the process. If the flow is unsteady, then the trajectories of the particles and the streamlines are different.

If the liquid is incompressible, then

$$\rho = const,$$

where  $\rho$  – the density of the liquid.

If the movement is stationary [18], then

$$\frac{\partial V}{\partial t} = 0.$$

For this model problem, the solution area is a square with dimensions  $x \in [0; 1], y \in [0; 1]$ .

The structure of neural networks allows us to calculate analytically the gradient of the error functional. This problem was solved in two ways: with analytical calculation of the gradient of the error functional and with the use of automatic differentiation. Analytical calculation allowed to speed up the work of the program. This article presents the results of the solution obtained

using the analytical calculation of the gradient of the error functional. Each variable had its own network of radial basis functions based on physics. All three networks are connected by a common root-mean-square error functional. The functional consists of the sum of the squares of the residuals. To simplify the expression, the multiplier  $\frac{1}{2}$  has been added for the gradient of the functional. The network was trained to the value of the total error functional equal to  $10^{-4}$ . The error functional can be written as

$$\begin{aligned}
I = & \frac{1}{2} \sum_{i=1}^N \left( u_i \frac{\partial u_i}{\partial x} + v_i \frac{\partial u_i}{\partial y} + \frac{\partial p_i}{\partial x} - \frac{1}{Re} \left( \frac{\partial^2 u_i}{\partial x^2} + \frac{\partial^2 u_i}{\partial y^2} \right) \right)^2 + \\
& \frac{1}{2} \sum_{i=1}^N \left( u_i \frac{\partial v_i}{\partial x} + v_i \frac{\partial v_i}{\partial y} + \frac{\partial p_i}{\partial y} - \frac{1}{Re} \left( \frac{\partial^2 v_i}{\partial x^2} + \frac{\partial^2 v_i}{\partial y^2} \right) \right)^2 + \\
& \frac{1}{2} \sum_{i=1}^N \left( \frac{\partial u_i}{\partial x} + \frac{\partial v_i}{\partial y} \right)^2 + \frac{\lambda}{2} \sum_{j=1}^K (u_j - u_j^A)^2 \\
& + \frac{\lambda}{2} \sum_{j=1}^K (v_j - v_j^A)^2 + \frac{\lambda}{2} \sum_{j=1}^K (p_j - p_j^A)^2
\end{aligned}$$

where  $Re$  – Reynolds number,  $N$  – number of internal test points,  $K$  – number of boundary test points,  $\lambda$  – penalty multiplier,  $u_j^A, v_j^A, p_j^A$  – analytical value of the  $j$ th test point at the boundary of the region.

The model is implemented in the MATLAB environment. The centers of the radial basis functions of each network were located on a uniform grid. The weight and width vectors were initialized with zero values. Experiments were conducted for each network to select the optimal number of neurons and trial points. The number of neurons for each physics-informed radial basis function networks is 64. The number of interior trial points for each network is 100. The number of boundary trial points for each network is 40. The coordinates of collocation points were generated as random numbers that were uniformly distributed in the solution domain. The network was trained using Nesterov's method. Only the weights were customizable. The location of the test points is shown in Figure 2.

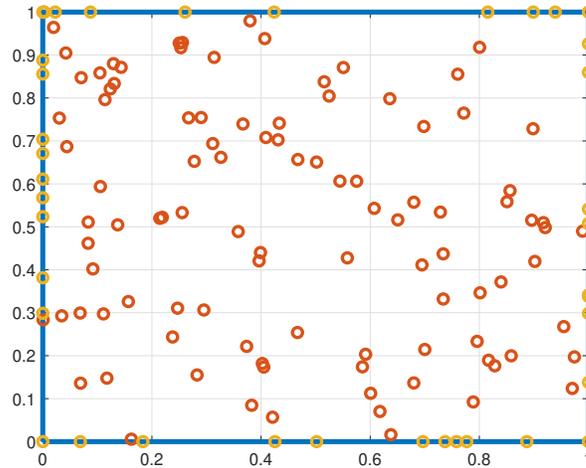


Fig. 2. Test points

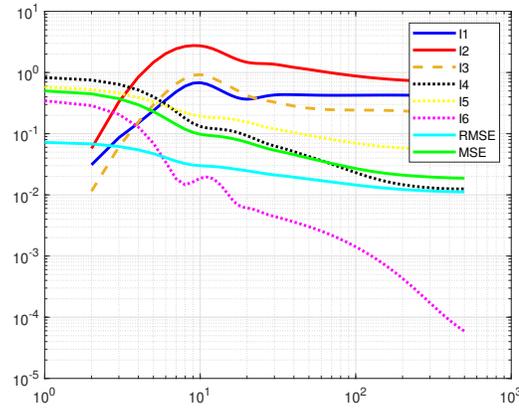
Let us analyze the effectiveness of the Nesterov method using the example of a model problem simulating the Kovaszny flow. The Nesterov method uses the history of parameter changes and the idea of the pulse accumulation method [16, 28]. The formula for calculating the parameters in this case takes the form

$$\Delta\Theta^{(k+1)} = \alpha\Delta\Theta^{(k)} - \gamma\text{grad}I(\Theta^{(k)} - \alpha\Delta\Theta^{(k)}).$$

where  $\Theta$  — the vector of one of the network parameters (we can consider the vector of all parameters),  $\gamma$  — the selected numerical coefficient (learning rate),  $\alpha$  — the moment coefficient, which takes values in the interval  $[-0.5; 1.5]$ .

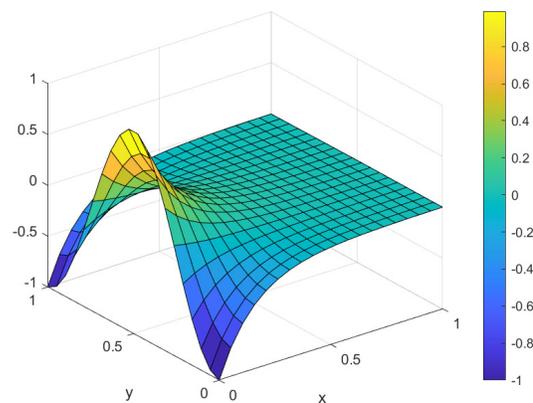
$I_1, I_2, I_3$  are the error values for the three Navier-Stokes equations, respectively.  $I_4, I_5, I_6$  are the error values for the three boundary conditions, respectively. The graph also shows the mean square error functional values (MSE) and the root mean square error functional values (RMSE). The root mean square error functional value of  $10^{-2}$  is achieved in on average of 150 iterations. This problem was solved on a fully connected network [25]. Physics-informed radial basis function networks are as accurate as fully connected networks, but they are easier and faster to train. Physics-informed radial basis function networks allowed us to achieve a root mean square error functional value of  $10^{-2}$  on average in 15 minutes.

Figure 3 shows the behavior of the error functional.



*Fig. 3.* Error functional

Plots of analytical solutions for velocity and pressure components are presented in Figures 4, 6, 8. The graphs obtained as a result of the network operation are shown in Figures 5, 7, 9. They have a visual similarity to the analytical solution. The value of the root mean square error functional for the first velocity component is on average  $10^{-6}$ . The value of the root mean square error functional for the second velocity component is on average  $10^{-6}$ . The value of the root mean square error functional for pressure averages  $10^{-4}$ . This is a testament to the effectiveness of the program. Error graphs compared with the analytical solution for the two components of velocity and pressure are shown in Figures 10, 11, 12.



*Fig. 4.* Analytical solution for the first velocity component

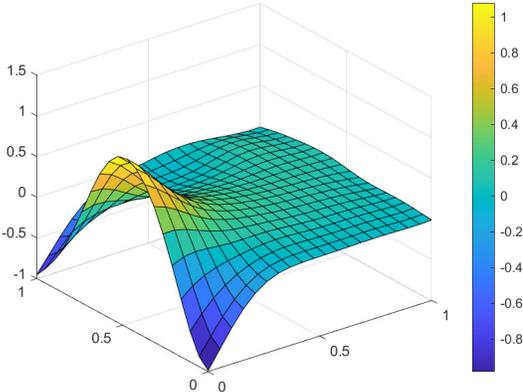


Fig. 5. The first component of speed resulting from network operation

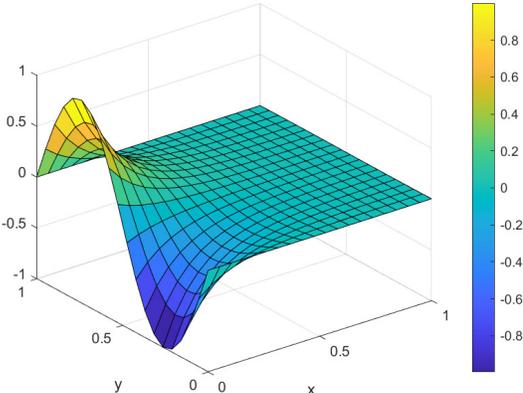


Fig. 6. Analytical solution for the second velocity component

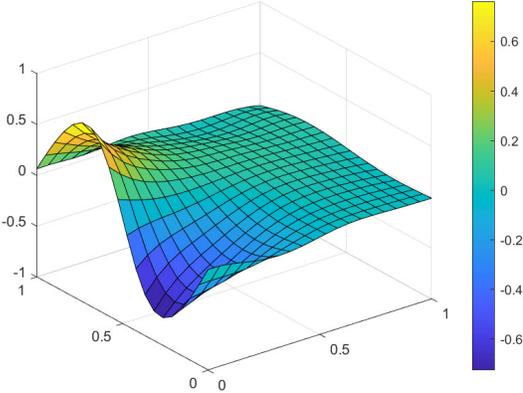


Fig. 7. The second component of speed obtained as a result of network operation

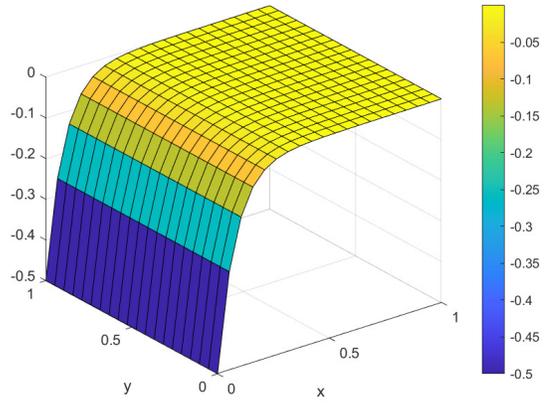


Fig. 8. Analytical solution for pressure

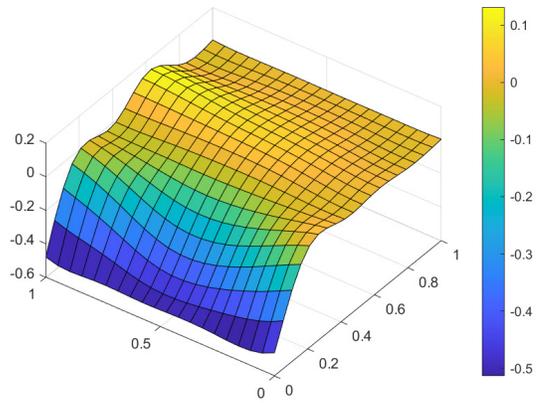


Fig. 9. Pressure obtained as a result of network operation

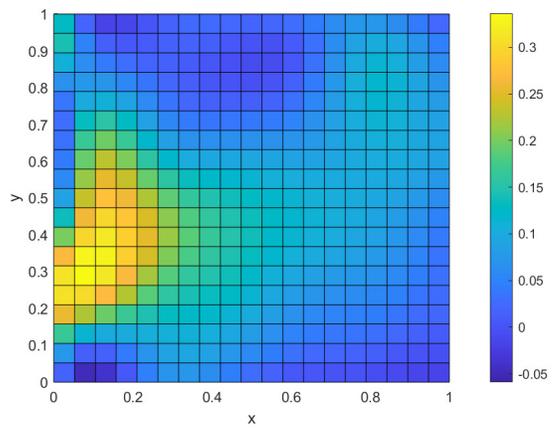
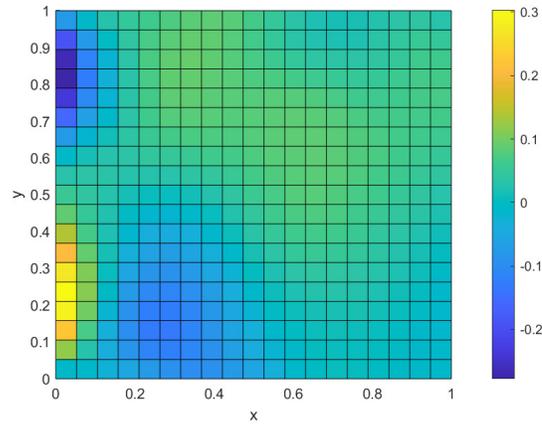
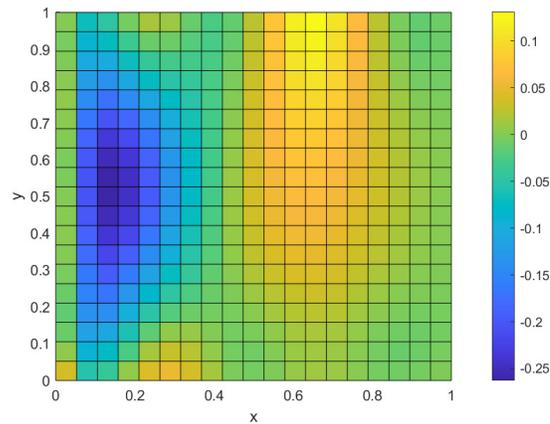


Fig. 10. Error compared to the analytical solution for the first speed component



*Fig. 11.* Error compared to the analytical solution for the second speed component



*Fig. 12.* Error compared to the analytical solution for pressure

## 4. Conclusion

Algorithms and programs for solving direct and inverse boundary value problems on physics-informed radial basis function networks have been developed. An algorithm and programs for solving a system of differential equations for solving two-dimensional stationary Navier-Stokes equations using physics-informed radial basis function networks have been developed. The gradient descent method, Nesterov's method, Levenberg-Marquardt method, and Hessian diagonal approximation method have been adapted to solve the tasks. Further prospect of the work is supposed to be developing an extension for one of the neural network implementation libraries in order to implement physics-informed radial basis function networks using automatic differentiation. This will expand the possibilities of using physics-informed radial basis function networks and allow objectively comparing physics-informed neural networks of different architectures.

## Declarations

**Data availability** This manuscript has no associated data.

**Ethical Conduct** Not applicable.

**Conflicts of interest** The authors declare that there is no conflict of interest.

## References

1. Buhmann M.D. *Radial basis functions: Theory and implementations*, Cambridge, Cambridge University Press. 2004. P. 259.
2. Cybenko G., *Approximation by superposition of a sigmoidal function. mathematics of control, Signals and Systems*. 1989. Vol. 2. P. 303–314.
3. Fotiadis D.I., Lagaris I.E., Likas A. *Artificial neural networks for solving ordinary and partial differential equations*, IEEE Transactions on Neural Networks. 1998. Vol. 9. P. 987–1000.
4. Giampaolo F., Rozza G., Raissi M., Francesco F., Piccialli S. Cuomo and V.S. Di Cola, *Scientific machine learning through physics-informed neural networks: Where we are and what's next* Journal of Scientific Computing. 2022. Vol. 92. P. 88.
5. Hecht-Nielsen R. *Kolmogorov's mapping neural network existence theorem* IEEE First Annual Int. Conf. on Neural Networks. 1987. Vol. 3, № 2. P. 11–13.
6. Hornik K., *Approximation capabilities of multilayer feedforward networks*, Neural Networks. 1999. Vol. 4, № 2. P. 251–257.
7. Karniadakis G.E., Raissi M., Perdikaris P. *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations* Journal of Computational Physics. 2019. Vol. 378. P. 686–707.
8. Kolmogorov A.N. *On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition* Russian Academy of Sciences. 1957. Vol. 114, № 5. P. 953–956.
9. Kovasznay L.S.G. *Laminar flow behind a two-dimensional grid*, Mathematical Proceedings of the Cambridge. 1948. Vol. 44, № 1. P. 58–62.
10. Kumar M., Yadav N., Yadav A., *An introduction to neural network methods for differential equations*, University Press, Oxford etc. 2015. P. 115.
11. Liu Z., Wang Y., Vaidya S., Ruehle F., Halverson J., Soljacic M., Hou T.Y., Tegmark M. *Kan: Kolmogorov-arnold networks* <https://arxiv.org/abs/2404.19756>.
12. Li Z. *Kolmogorov-arnold networks are radial basis function networks* <https://arxiv.org/abs/2405.06721>.

13. Mao Z., Karniadakis G.E., Lu L., Meng X. *Deepxde: A deep learning library for solving differential equations* SIAM Review. 2021. Vol. 63, № 1. P. 208–228.
14. Marquardt D.W., *An algorithm for least-squares estimation of nonlinear parameters*, Journal of the Society for Industrial and Applied Mathematics. 1963. Vol. 11. P. 431–441.
15. Morozov V.A. *Methods for solving incorrectly posed problems* New York: Springer. 1984. P. 280.
16. Nesterov Y. E. *Introduction to convex optimization* Moscow Center for Continuous Mathematical Education. 2010. P. 280.
17. Ramachandran P., Ramabathiran A.A. *Spinn: Sparse, physics-based, and partially interpretable neural networks for pdes* Journal of Computational Physics. 2021. P. 445.
18. Slezkin N. A. *Dynamics of viscous incompressible fluid* state publishing house of technical and theoretical literature. 1955.
19. Stenkin D.A., Gorbachenko V.I. *Deep radial basis function networks*, 7th International Conference on Contemporary Information Technology and Mathematics. 2021. P. 267–273.
20. Stenkin D.A., Gorbachenko V.I. *Mathematical modeling on a physics-informed radial basis function network*, Mathematics. 2024. Vol. 12, № 2. P. 241.
21. Stenkin D.A., Gorbachenko V.I. *Physics-informed radial basis function networks: Solving inverse problems for partial differential equations*, Cyber-Physical Systems and Control II. CPSC 2021. Lecture Notes in Networks and Systems. 2023. Vol. 460. P. 3–12.
22. Stenkin D.A., Gorbachenko V.I. *Physics-informed radial basis-function networks*, Technical Physics. 2023. Vol. 68. P. 151–157.
23. Stenkin D.A., Gorbachenko V.I. *Solving equations describing processes in a piecewise homogeneous medium on radial basis functions networks* Neuroinformatics 2020. Studies in Computational Intelligence. 2021. Vol. 1008. P. 412–419.
24. Stenkin D.A., Gorbachenko V.I. *Solving of inverse coefficient problems on networks of radial basis functions*, Advances in Neural Computation, Machine Learning, and Cognitive Research V. Neuroinformatics 2021. 2022. Vol. 1008. P. 230–237.
25. Strizhak S.V., Koshelev K.B. *The use of a physically based neural network on the example of modeling hydrodynamic processes that allow an analytical solution* Proceedings of the Institute of System Programming of the Russian Academy of Sciences. 2023. Vol. 35, № 5. P. 245–258.
26. Vabishchevich P.N., Samarskii A.A. *Numerical methods for solving inverse problems of mathematical physics* Berlin: Walter de Gruyter. 2007. Vol. 52. P. 454.
27. Wong A., Hryniewski A. *Deeplabnet: End-to-end learning of deep radial basis networks with fully learnable basis functions* <https://arxiv.org/abs/1911.09257>
28. Wright S., Nocedal J. *Numerical optimization* Springer. 2006. Vol. 378.

