



УДК 004.032.26, 004.822.2
<https://doi.org/10.37661/1816-0301-2023-20-3-90-105>

Оригинальная статья
Original Paper

Семантические модели и средства разработки искусственных нейронных сетей и их интеграции с базами знаний

М. В. Ковалёв

*Белорусский государственный университет
информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь
E-mail: kovalev@bsuir.by*

Аннотация

Цели. Предлагаются спецификации моделей и средств разработки искусственных нейронных сетей (ИНС) и их интеграции с базами знаний интеллектуальных систем. Актуальность исследования определяется необходимостью решения комплексных задач интеллектуальными системами, алгоритм и методы решения которых отсутствуют в базах знаний интеллектуальных систем.

Методы. Анализируются четыре уровня интеграции ИНС с базами знаний. В ходе анализа формулируются требования и спецификации к необходимым моделям и средствам разработки и интеграции. Специфицированные на каждом уровне модели и средства включают в себя модели и средства предыдущего уровня. Применение средств рассмотрено на примере решения задачи классификации сущностей базы знаний с помощью графовой нейронной сети.

Результаты. Разработаны спецификации модели представления ИНС в базе знаний, агентно-ориентированной модели разработки и интерпретации ИНС, обеспечивающие интеграцию ИНС в базы знаний на всех выделенных уровнях, а также метод классификации сущностей базы знаний с помощью графовой нейронной сети.

Заключение. Предложенные модели и средства позволяют интегрировать в базу знаний интеллектуальной системы любые обученные ИНС и использовать их для решения комплексных задач в рамках технологии OSTIS. Становятся возможными проектирование и обучение ИНС на основании как внешних данных, так и фрагментов базы знаний, а также автоматизация процесса разработки ИНС в базе знаний.

Ключевые слова: интеграция, методы решения задач, искусственные нейронные сети, графовые нейронные сети, базы знаний, онтологии

Для цитирования. Ковалёв, М. В. Семантические модели и средства разработки искусственных нейронных сетей и их интеграции с базами знаний / М. В. Ковалёв // Информатика. – 2023. – Т. 20, № 3. – С. 90–105. <https://doi.org/10.37661/1816-0301-2023-20-3-90-105>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 12.07.2023

Подписана в печать | Accepted 01.08.2023

Опубликована | Published 29.09.2023

Semantic models and tools for the development of artificial neural networks and their integration into knowledge bases

Mikhail V. Kovalev

*Belarusian State University of Informatics and Radioelectronics,
st. P. Brovki, 6, Minsk, 220013, Belarus
E-mail: kovalev@bsuir.by*

Abstract

Objectives. Specifications of models and tools for the development of artificial neural networks (ANNs) and their integration into knowledge bases (KBs) of intelligent systems are being developed. The relevance is determined by the necessity of implementing the possibility to solve complex problems by intelligent systems, which algorithms and methods of solving are not available in the knowledge base of the intelligent system.

Methods. Four levels of integration of artificial neural networks into knowledge bases are formulated and analyzed. During the analysis the requirements and specifications for required models and tools for the development and integration are formulated. Specified at each level the models and tools include the models and tools of previous level. The application of the tools is considered by the example of solving the problem of classifying the knowledge base entities using a graph neural network.

Results. The specifications of the ANN representation model in the knowledge base, the agent-based model for the development and interpretation of the ANN, which ensures the integration of the ANN into knowledge bases at all selected levels, as well as the method for classifying knowledge base entities using a graph neural network, have been developed.

Conclusion. The developed models and tools allow integrating any trained ANNs into the knowledge base of the intelligent system and using them to solve complex problems within the framework of OSTIS technology. It also becomes possible to design and train ANNs both on the basis of external data and on the basis of fragments of the knowledge base. Automation of ANNs development process in the knowledge base becomes available.

Keywords: integration, problem-solving methods, artificial neural networks, graph neural networks, knowledge bases, ontologies

For citation. Kovalev M. V. *Semantic models and tools for the development of artificial neural networks and their integration into knowledge bases*. *Informatika [Informatics]*, 2023, vol. 20, no. 3, pp. 90–105 (In Russ.). <https://doi.org/10.37661/1816-0301-2023-20-3-90-105>

Conflict of interest. The author declares no conflict of interest.

Введение. В настоящее время продолжается активное совершенствование отдельных интеллектуальных методов решения задач. Отлично показали себя большие языковые модели [1, 2], генеративные нейронные сети [3], технологии построения и внедрения графов знаний [4]. Однако с развитием искусственного интеллекта (ИИ) становится все более явной потребность в эффективной интеграции различных методов решения задач. Это вызвано необходимостью создания более гибких и адаптивных интеллектуальных систем, способных решать сложные задачи и эффективно использовать имеющиеся знания и данные. Такие системы должны использовать интеллектуальные методы решения задач в самых различных комбинациях и последовательностях, не всегда заранее известных проектировщикам интеллектуальных систем. Появились целые междисциплинарные подходы [5, 6], призванные объединить в себе преимущества двух традиционных направлений ИИ: коннекционистского и символического.

Одной из основных проблем, стоящих перед исследователями и разработчиками в области ИИ, является определение оптимальных стратегий интеграции различных методов решения задач, как интеллектуальных, так и традиционных. Обилие комплексных задач (задач, решение которых невозможно с помощью одной модели решения задач, одного вида знаний, одной интеллектуальной компьютерной системы) порождает многообразие архитектур интеллектуальных систем, призванных решать задачи из какого-то ограниченного набора классов задач с помощью фиксированного набора методов их решения. Отсутствие унификации подходов к интеграции методов решения задач в интеллектуальные системы делает затруднительным повтор-

ное использование этих методов в других интеллектуальных системах. Отдельные же комплексные задачи [7] в принципе не могут быть решены без унифицированного подхода к интеграции и интерпретации методов решения задач. Именно такими задачами являются: проектирование ИНС, обучение и дообучение ИНС, верификация работы ИНС в реальном времени на основании описания задачи и ее контекста.

Целью настоящего исследования является спецификация моделей и средств разработки (проектирования и обучения) и интеграции ИНС с базами знаний (БЗ) интеллектуальных систем. Под спецификацией сущности подразумевается описание синтаксической, денотационной и операционной семантики понятий, необходимых для формализации данной сущности в БЗ, а также связи этой сущности с другими сущностями. Обоснование того, почему речь идет об интеграции с БЗ, будет дано во втором разделе. В соответствии с поставленной целью можно выделить следующие задачи исследования:

- анализ современных подходов к интеграции интеллектуальных методов решения задач с БЗ;
- формулирование требований к моделям и средствам разработки ИНС и их интеграции с БЗ;
- специфицирование моделей разработки ИНС и их интеграции с БЗ;
- специфицирование средств разработки ИНС и их интеграции с БЗ.

Исследование построено следующим образом. Выделены четыре уровня интеграции ИНС, каждый уровень включает в себя предыдущий. При описании каждого уровня уточняются спецификации целевых моделей и средств. Выделены следующие уровни:

- ручной интеграции;
- интеграции на основании унификации интерфейсов использования;
- интеграции на основании унификации интерфейсов построения;
- конвергенции.

На последнем уровне специфицированы целевые модели и средства, дающие наиболее глубокий уровень интеграции и покрывающие наибольшее количество классов задач, решаемых интеллектуальными системами с помощью ИНС.

1. Уровень ручной интеграции. Большинство современных интеллектуальных систем интегрируют различные интеллектуальные методы решения задач, в том числе ИНС, как черные ящики, разрабатывая специализированные интерфейсы и средства взаимодействия с каждым отдельным методом.

До недавнего времени интеллектуальные системы интегрировали в себе отдельные методы ИИ для решения узкого круга задач. Зачастую подобные интеграции сводятся к подключению некоторых ИИ-модулей к традиционным системам. Примером такой интеграции является добавление ИИ чат-ботов [8], модулей распознавания и синтеза речи [9], моделей компьютерного зрения [10] и т. д. Между тем в последние годы исследуются и разрабатываются системы, способные интегрировать в себе несколько методов ИИ в одной системе. В первую очередь речь идет о системах, интегрирующих в себе ИНС и БЗ, которые в зарубежных публикациях все чаще называются графами знаний [4].

Можно выделить следующие подходы к подобным интеграциям:

1. ИНС как сенсор интеллектуальной системы, основанной на БЗ [11, 12]. При данном подходе ИНС извлекают информацию из входных данных (текста, звука, изображения) и транслируют ее в БЗ, где эта информация используется для решения задач.

2. В БЗ интеллектуальной системы происходят обработка и верификация обучающих выборок и выходных данных ИНС [13, 14]. В современных задачах, решаемых с помощью ИНС, постоянно растут требования к уровню учета семантики в решении. Принимая во внимание особенности ИНС, уровень учета семантики поднимается за счет поиска логических и семантических противоречий в обучающих и выходных данных ИНС. В случае выявления подобных противоречий в выходных данных в процессе обучения ИНС становится возможной корректировка методов обучения ИНС.

3. ИНС обрабатывают и генерируют фрагменты БЗ. Активно развивается подход, получивший серьезный толчок к развитию благодаря внедрению графовых ИНС, способных принимать на вход графовые структуры. С помощью таких сетей предсказываются отношения в БЗ [15], классифицируются и кластеризуются фрагменты БЗ [16].

4. В БЗ хранятся логические правила, которые соответствуют конкретным нейронным элементам или слоям ИНС [17, 18]. Данный подход называется объясняющим ИИ (explainable AI) и призван решить классическую проблему ИНС – проблему «черного ящика» [19]. ИНС обучаются на исторических данных, они не способны объяснить, почему выдали тот или иной результат. Даже нашумевшие в последнее время большие языковые модели не способны объяснить, почему они выдали тот или иной результат [20], и даже если попросить их это сделать, они выдадут правдоподобные, но не истинные причины своих результатов. Суть подхода «объясняющий ИИ» к решению этой проблемы состоит в попытке сгенерировать правдоподобные логические правила, которые ставятся в соответствие этапам вычисления внутри ИНС. При работе ИНС оцениваются те этапы вычислений, которые наибольшим образом повлияли на результат, и строится цепочка логических правил, объясняющая его.

Все задачи, решаемые перечисленными подходами, безусловно, актуальны, однако зачастую их решения являются узконаправленными, сосредоточенными на конкретной проблеме из некоторой предметной области. Нейросетевые и семантические методы решения задач используются в четко заданной последовательности, вручную интегрируются между собой и не нацелены на решение комплексных задач произвольной формулировки. На рис. 1 показан пример схемы взаимодействия ИИ-модулей, интегрированных с решателем задач на ручном уровне.



Рис. 1. Схема взаимодействия решателя задач с ИИ-модулями на уровне ручной интеграции

Fig. 1. Scheme of interaction of the problem solver with AI modules at the level of manual integration

2. Уровень интеграции на основании унификации интерфейсов использования. Решение комплексных задач подразумевает:

1. Автоматическое разбиение задачи на подзадачи. Задается декларативная формулировка комплексной задачи, на основании которой системе необходимо выделить декларативную формулировку подзадач.

2. Поиск методов решения каждой подзадачи. В системе хранятся спецификации различных методов решения задач, в которых описаны класс решаемых задач, тип входных данных и т. п. На основании этих спецификаций система подбирает метод решения каждой подзадачи.

3. Использование найденных методов. Успешное решение всех подзадач обеспечивает успешное решение исходной задачи.

4. Описание решения исходной задачи как последовательности решения подзадач.

Исходя из описанных шагов, можно выделить минимальные требования к системам, обеспечивающим интеграцию различных методов решения задач:

- наличие унификации представления и согласованность различных видов знаний, описывающих задачи, контекст задач и методы их решения;
- наличие моделей и средств по разбиению задач на подзадачи;

– способность объяснить свои решения на уровне описания формулировок подзадач и последовательности их решения.

Интересный подход к решению комплексных задач и интеграции в себе различных методов их решения представлен в проекте ChatGPT (URL: <https://openai.com/blog/chatgpt>) и его системе плагинов (URL: <https://openai.com/blog/chatgpt-plugins>). ChatGPT является большой языковой моделью, способной решать комплексные задачи, формулировки и контекст которых сформулированы на естественном языке. Основным методом решения этих задач является сама большая языковая модель, однако система плагинов к данной модели позволяет ей использовать различные методы решения задач.

Плагин к ChatGPT представляет собой открытый сервис, предоставляющий некоторое API. Спецификация API задается в структурированном формате, в котором описываются типы входных данных для каждой точки доступа и ее суть на естественном языке (URL: <https://platform.openai.com/docs/plugins/getting-started/openapi-definition>). Когда пользователь задает вопрос, модель может выбрать вызов API из некоторого плагина, если спецификация этого API близка к контексту решаемой задачи. Вызов API пополняет контекст решаемой задачи и повышает уровень вероятности того, что сгенерированный ответ решит задачу пользователя. К примеру, плагин может возвращать ссылки на интернет-источники (URL: <https://gptstore.ai/plugins/-simbiss-net>), чего по умолчанию сама модель делать не может.

Так как формулировка задачи, контекст и описание методов ее решения частично задаются на естественном языке, то можно говорить о выполнении первого из трех сформулированных требований. С небольшими оговорками можно говорить о выполнении и второго требования, так как модель пополняет контекст поставленной задачи на основании разбиения формулировки поставленной задачи на подзадачи. К примеру, из запроса «Найди ссылку на исследование по ИНС и дай ему свою оценку» система сможет выделить ключевое слово «ссылка» и воспользоваться плагином по поиску ссылок в Интернете.

Между тем на данный момент модель не может объяснить, какие подзадачи были выделены и как их решение повлияло на решение исходной задачи. Плагины пополняют контекст решаемой задачи, который отправляется на вход большой языковой модели, объяснить ответ которой уже нельзя. В результате можно говорить о неудовлетворении последнего требования.

Другой подход к интеграции различных методов решения задач основан на формальном представлении знаний, описывающих контекст задач и методы их решения. Данный подход лежит в основе технологии OSTIS [21], с помощью которой проектируются и реализуются ostis-системы – интеллектуальные системы, решающие комплексные задачи на основании унификации знаний и методов решения задач.

В качестве формальной основы для представления знаний в рамках технологии OSTIS используется унифицированная семантическая сеть с теоретико-множественной интерпретацией. Такая модель представления названа SC-кодом (semantic computer code) [21]. Элементы семантической сети названы sc-узлами и sc-коннекторами (sc-дугами, sc-ребрами).

Любая ostis-система состоит из БЗ, решателя задач и пользовательского интерфейса (рис. 2).



Рис. 2. Архитектура ostis-системы

Fig. 2. Architecture of the ostis-system

Основу БЗ, построенной по технологии OSTIS, составляет иерархическая система предметных областей (ПрО) и соответствующих им онтологий. Онтологии содержат описания понятий, необходимые для формализации знаний из соответствующей этой онтологии ПрО. Любое знание, описывающее некоторую задачу, ее контекст и спецификацию методов решения, можно представить в виде конструкций SC-кода. Таким образом, обеспечивается унификация представления и согласованность различных видов знаний, описывающих задачи, их контекст и методы решения.

Решатель задач занимается обработкой фрагментов БЗ. На операционном уровне обработка сводится к добавлению, поиску, редактированию и удалению sc-узлов и sc-коннекторов БЗ. На семантическом уровне такая операция является действием, выполняемым в памяти субъекта действия, где в общем случае субъектом служит ostis-система, а БЗ – ее памятью. Действие определяется как процесс воздействия одной сущности (или некоторого множества сущностей) на другую сущность (или на некоторое множество других сущностей) согласно некоторой цели.

Действия выполняются агентами в соответствии с поставленными задачами. Задача – это формальная спецификация некоторого действия, достаточная его для выполнения каким-либо субъектом. В зависимости от конкретного класса задач описываться может как внутреннее состояние самой интеллектуальной системы, так и требуемое состояние внешней среды [21]. Схожие задачи объединены в классы, для которых заданы обобщенные формулировки задач. В работе [22] приведены классы задач, решаемых ИНС.

Решатель задач ostis-системы имеет модели и средства по разбиению задач на подзадачи и способен объяснить свои решения на уровне описания формулировок подзадачи и последовательности их решения. При решении задачи решатель задач разбивает ее на подзадачи с явным описанием формулировки каждой подзадачи, ищет метод ее решения и применяет его.

Таким образом, системы, построенные по принципам технологии OSTIS, удовлетворяют всем трем выдвинутым требованиям. Поэтому в рамках предлагаемого подхода описываются модели и средства разработки ИНС и их интеграции в БЗ с помощью технологии OSTIS.

В соответствии с архитектурой ostis-систем для интеграции ИНС в интеллектуальную систему необходимо описать:

- модель представления ИНС в БЗ;
- агентно-ориентированную модель разработки и интерпретации ИНС;
- библиотеку многократно используемых компонентов по разработке и интерпретации ИНС, а также библиотеку уже разработанных ИНС.

С точки зрения ostis-систем ИНС является методом решения задач. Метод решения задач определяется как программа решения задач соответствующего класса, которая может быть и процедурной, и декларативной. В свою очередь, класс методов решения задач определяется как множество всевозможных методов решения задач, имеющих общий язык представления этих методов. Язык представления методов позволяет описывать синтаксическую, денотационную и операционную семантику этих методов. Для того чтобы добавить в арсенал технологии OSTIS такой метод решения задач, как ИНС, и тем самым расширить круг задач, решаемых ostis-системами, необходимо описать синтаксис, денотационную и операционную семантику языка представления ИНС. Синтаксис языка представления ИНС является синтаксисом SC-кода, использующимся в технологии OSTIS для представления знаний.

Денотационная семантика языка представления ИНС описывается в рамках модели представления ИНС в БЗ, которая сводится к описанию ПрО и соответствующей ей онтологии ИНС.

На уровне интеграции по входам и выходам в рамках данной ПрО содержатся те знания о готовых к использованию ИНС, которые необходимы решателю задач для принятия решения об использовании той или иной ИНС. Такими знаниями являются:

- тип входных данных;
- множество распознаваемых классов или тип выходных данных;
- класс задач, решаемых ИНС;
- архитектура;

- среднее время работы;
- метрики эффективности.

Соответствующие понятия уже определены в рамках работы [22]. На рис. 3 изображена частичная спецификация обученной ИНС [12], описанной с помощью предлагаемой модели представления ИНС. Данная ИНС построена на основании архитектуры eXnet [23] и решает задачу классификации эмоций.

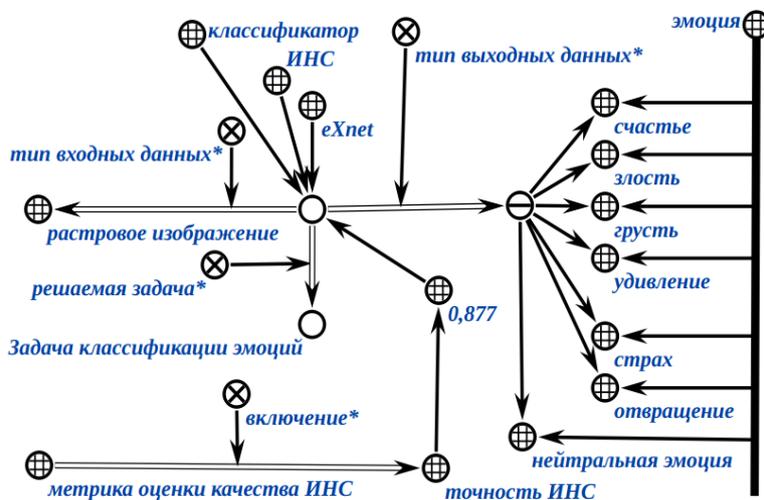


Рис. 3. Частичная спецификация обученной ИНС по распознаванию эмоций

Fig. 3. Partial specification of a trained ANN for emotion recognition

Операционной семантикой любого языка представления методов решения задач является спецификация семейства агентов, обеспечивающих интерпретацию (использование) любого метода, принадлежащего соответствующему классу методов. Это семейство служит интерпретатором соответствующего метода решения задач. В рамках технологии OSTIS такой интерпретатор называется моделью решения задач. Так как в рамках технологии OSTIS используется многоагентный подход, то разработка модели решения задач с помощью ИНС сводится к разработке агентно-ориентированной модели разработки и интерпретации ИНС. Согласно многоагентному подходу каждый агент выполняет некоторое действие в sc-памяти. Следовательно, модель задается иерархией специфицированных действий и соответствующей этим действиям иерархией агентов. На рассматриваемом уровне интеграции можно выделить только одно действие – по интерпретации ИНС, интегрированной по входам и выходам. Такому действию соответствует семейство агентов, каждый из которых вызывается решателем задач в случае выбора некоторой описанной в БЗ ИНС, представленной на том сервисе, за интеграцию с которым отвечает данный агент. Вызываемому агенту передается узел, обозначающий соответствующую ИНС, и входные данные, тип которых определяется по описанию ИНС.

Все агенты, реализующие операционную семантику языка представления ИНС, а также конкретные ИНС, описанные и интегрированные в ostis-систему, являются частью библиотеки многократно используемых компонентов по работе с ИНС.

Описанные модели и средства будут уточняться по мере рассмотрения более глубоких уровней интеграции.

3. Уровень интеграции на основании унификации интерфейсов построения. Данный уровень интеграции позволяет не только интерпретировать готовые ИНС, но и выполнять такие действия по разработке ИНС, как проектирование и обучение.

В памяти ostis-системы хранятся знания о том, ИНС какого класса могут решить задачу заданного класса, но необходимый экземпляр класса ИНС может не быть представлен в системе. В этом случае система должна иметь возможность сообщить пользователю о необходимости разработки ИНС для решения поставленной задачи. Так как система хранит в единой памяти

задачу и требования к методу ее решения, появляется возможность разработать необходимый метод задачи решения.

В итоге становится возможным в ходе решения некоторой задачи не только использовать готовые ИНС, но и проектировать и обучать специализированные ИНС для решения текущей задачи. Такие проектирование и обучение могут осуществляться системой самостоятельно, когда в БЗ описаны либо все необходимые для этого знания (выборка, критерии обучения, спецификация входов и выходов и т. д.), либо источники подобных знаний, к которым система может обращаться самостоятельно. В случае когда таких возможностей нет, система может обращаться к разработчикам для пополнения БЗ необходимыми знаниями. К примеру, для решения задачи система может попросить разработчика предоставить обучающую выборку в требуемом формате.

На рассматриваемом уровне интеграции появляется принципиально новый класс задач по обучению ИНС обработке фрагментов БЗ. Соответственно, обучающая выборка также может состояться из фрагментов БЗ. В рамках исследования данного класса задач был разработан метод классификации сущностей БЗ. Постановка задачи сформулирована следующим образом: классифицировать сущности БЗ на основании их семантической окрестности. Метод описывается на примере классификации понятий онтологий. Такая задача является одной из подзадач задачи автоматического пополнения построения онтологий для диалоговой системы, управляемой знаниями и поддерживающей диалог на свободную тему [24]. Основным требованием к диалоговой системе является необходимость поддерживать диалог в рамках любой предметной области, а не только в рамках тех, которые предварительно формализованы разработчиками. К примеру, когда пользователь сообщил системе: «Я умею разрабатывать системы на C++», то система должна протранслировать эти знания, несмотря на то, что изначально в ее БЗ не содержалось предметных областей и онтологий о программировании. Таким образом, появилась необходимость решения задачи автоматического построения онтологий.

Выяснилось, что большие языковые модели при правильной настройке могут неплохо выделять смысл сообщений в виде троек: <пользователь, разрабатывать, система>, <пользователь, разрабатывать на, C++>, <C++, есть, язык программирования>, <разрабатывать, есть, навык> и т. д. Однако для того чтобы протранслировать такие тройки в SC-код, нужно выделить из них понятия и экземпляры. Если понятий еще нет в БЗ, то нужно их классифицировать, чтобы понять, к какой онтологии их отнести и каким образом использовать для формализации сообщений пользователя в будущем. К примеру, понятие «язык программирования» является абсолютным и обозначает множество, содержащее известные системе языки программирования, а понятие «разрабатывать» является неролевым отношением, связывающим разработчика и систему.

Выделяются следующие целевые классы:

- абсолютное понятие (`sc_node_not_relation`);
- ролевое отношение (`sc_node_role_relation`);
- неролевое отношение (`sc_node_norole_relation`).

В качестве БЗ использовалась мета-система OSTIS (URL: <https://ims.ostis.net>).

На рис. 4 представлена семантическая окрестность понятий, принадлежащих целевым классам.

Для решения задачи классификации сущностей в БЗ использовалась графовая нейронная сеть (Graph Neural Network, GNN) CapsGNN [25]. GNN – это семейство нейронных сетей, которые могут естественным образом работать с данными, структурированными на основе графа. Каждому элементу графа (вершине или ребру) ставится в соответствие некоторый эмбединг – вектор фиксированной размерности, характеризующий элемент графа. Каждый слой GNN преобразует эти эмбединги по некоторым правилам, которые учитывают эмбединги соседних элементов графа. Из эмбедингов элементов графа высчитывается эмбединг всего графа.

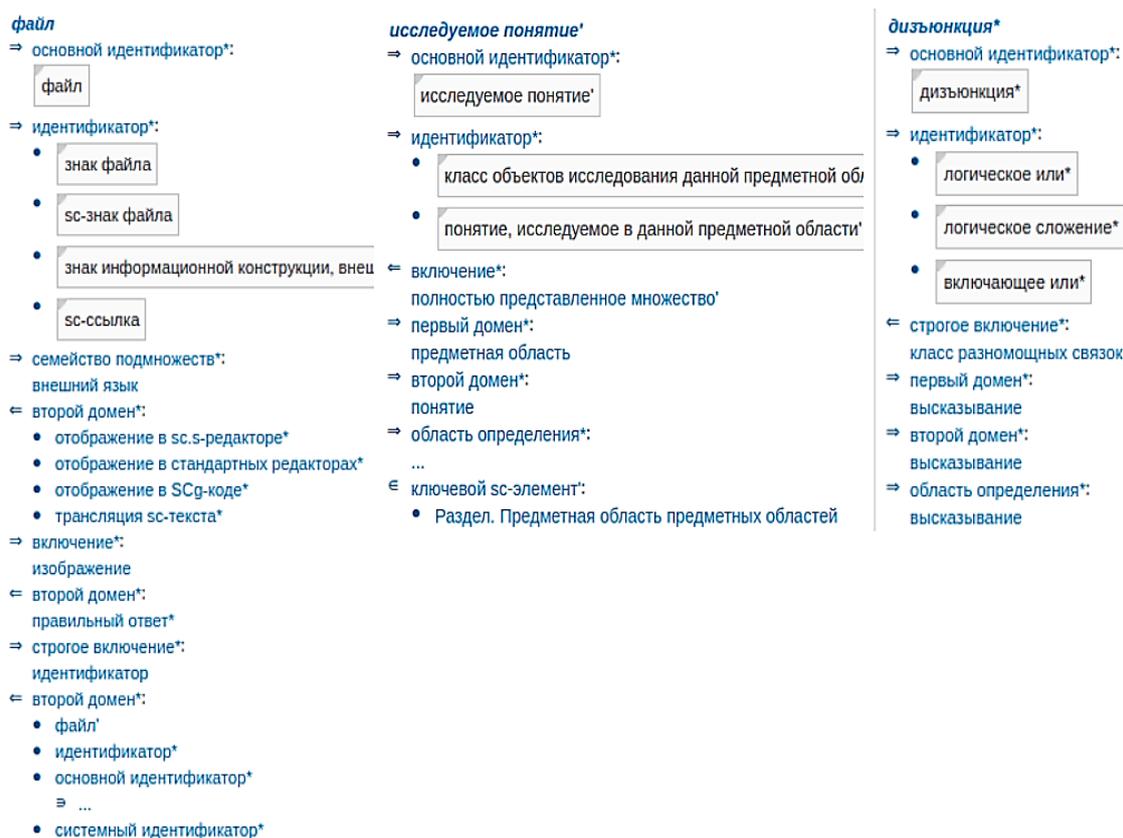


Рис. 4. Семантическая окрестность понятий целевых классов

Fig. 4. Semantic neighborhood of target class concepts

SC-код не является классическим графом, на работу с которым ориентирована сеть CapsGNN. С точки зрения теории графов SC-код – это неклассический граф, в котором началом и концом дуг (ребер) могут выступать не только вершины, но и другие дуги (ребра). Поэтому был разработан алгоритм преобразования SC-кода в классический граф, суть которого состоит в замене инцидентности между дугами на вершину графа. На рис. 5 показан пример преобразования фрагмента SC-кода в классический граф в несколько этапов:

- в ходе преобразования состояния 1 в 2 выполняется замена общей дуги на связку с явным указанием первого и второго элемента этой связки;
- в ходе преобразования состояния 2 в 3 выполняется удаление инцидентности между дугами путем ввода новой вершины графа в месте инцидентности дуг;
- в ходе преобразования состояния 3 в 4 выполняются нумерация вершин полученного графа и подстановка в соответствие каждой вершине вектора с одним элементом – номером типа sc-элемента.

В ходе исследования в качестве начального эмбединга вершин получившегося графа рассматривались следующие характеристики:

- количество исходящих дуг;
- количество входящих дуг;
- расстояние до понятий онтологии верхнего уровня;
- номер типа sc-элемента (в зависимости от того, является ли sc-элемент узлом или дугой, обозначает отношение, класс, структур, связку или что-то еще, sc-элементу присваиваются разные номера);
- различные сочетания вышперечисленного.

В результате перебора наилучшим классифицирующим признаком был выбран номер типа sc-элемента.

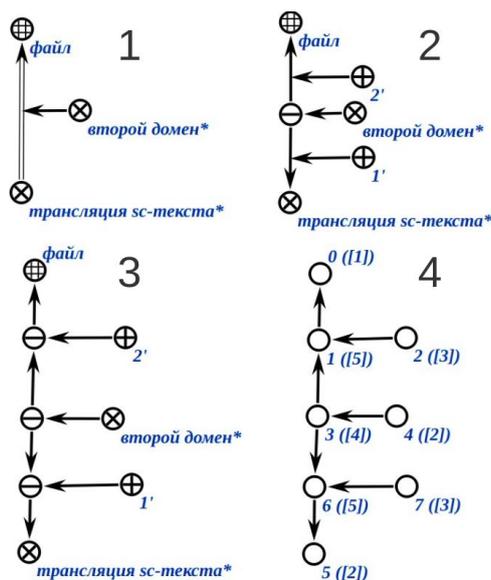


Рис. 5. Преобразование фрагмента SC-кода в классический граф

Fig. 5. Converting a fragment of SC-code into a classical graph

В рамках исследования также решалась задача выбора той части семантической окрестности понятия, которая наиболее характерна для класса данного понятия. В качестве такой части рассматривались:

- множества, в которых состоит понятие, с указанием его роли;
- элементы, которые принадлежат понятию, с указанием его роли;
- исходящие неролевые отношения с другими sc-элементами;
- входящие неролевые отношения с другими sc-элементами;
- различные сочетания вышеперечисленного.

Наилучшие результаты были получены при использовании той части семантической окрестности понятия, которая описывает входящие неролевые отношения с другими sc-элементами.

В таблице представлены результаты обучения CapsGNN, их можно назвать приемлемыми для решения поставленной задачи. Точность обучения отображается отношением количества правильно классифицированных элементов тестовой выборки к количеству всех элементов этой выборки.

Результаты обучения CapsGNN

CapsGNN training results

Количество эпох <i>Number of epochs</i>	Размер выборки <i>Sample size</i>		Среднее количество <i>Average quantity</i>		Точность <i>Accuracy</i>
	обучающей <i>teaching</i>	тестовой <i>test</i>	вершин <i>peaks</i>	дуг <i>arcs</i>	
100	65	28	19	11	0,89
200	65	28	19	11	0,92
300	65	28	19	11	0,92

Рассматриваемую задачу можно обобщить до следующей формулировки: определить принадлежность sc-элемента одному из заданных множеств.

При реализации рассматриваемого уровня интеграции решатель задач автоматически выделит и решит следующие подзадачи:

- поиск класса метода решения поставленной задачи с помощью GNN;
- составление обучающей выборки в БЗ;

- выбор подходящей части семантической окрестности элементов;
- вычисление начальных эмбедингов элементов;
- обучение GNN классификации по заданным множествам;
- решение задачи с помощью новой ИНС.

Основным преимуществом интеграции ИНС на основании унификации интерфейсов разработки является возможность решать задачи с помощью методов, которые не были известны системе на момент формулировки задач. Система начинает обретать навыки для решения поставленной задачи, что является важным шагом на пути к созданию сильного ИИ.

4. Уровень конвергенции. Все предыдущие уровни интеграции все еще не решают проблему черного ящика, так как ИНС проектируются, обучаются и интерпретируются в сторонних по отношению к БЗ модулях или сервисах. В БЗ известно только то, какие входы были переданы в ИНС и какие выходы были получены. Однако процесс преобразования входов в выходы все еще остается недоступным для формального описания внутри БЗ и анализа этого процесса с помощью имеющихся у интеллектуальной системы методов.

Уровень конвергенции подразумевает не только описание интерфейсов, но и выполнение процессов проектирования, обучения и интерпретации ИНС в БЗ. Перенос всех этих процессов в БЗ позволяет:

- анализировать процесс преобразования входных данных в выходные в ходе интерпретации ИНС с помощью других интегрированных в БЗ методов решения задач;
- оказывать интеллектуальную поддержку разработчикам ИНС;
- обеспечивать платформенную независимость *ostis*-систем, использующих ИНС.

Для интерпретации любого метода решения задач в БЗ необходимо иметь достаточный уровень описания денотационной и операционной семантики языка представления этого метода.

На уровне конвергенции денотационная семантика языка представления ИНС уточняется такими понятиями онтологии ИНС, как *нейрон**, *распределительный слой**, *обработывающий слой**, *функция активации**, *весовой коэффициент* и т. д. Полный набор понятий дан в работе [22], продолжением которой является настоящая статья. На рис. 6 показан пример формализации в БЗ полносвязной двухслойной ИНС с двумя нейронами на входном слое и одним нейроном на обработывающем слое. Формализованную ИНС интерпретируют специализированные агенты для решения задач.

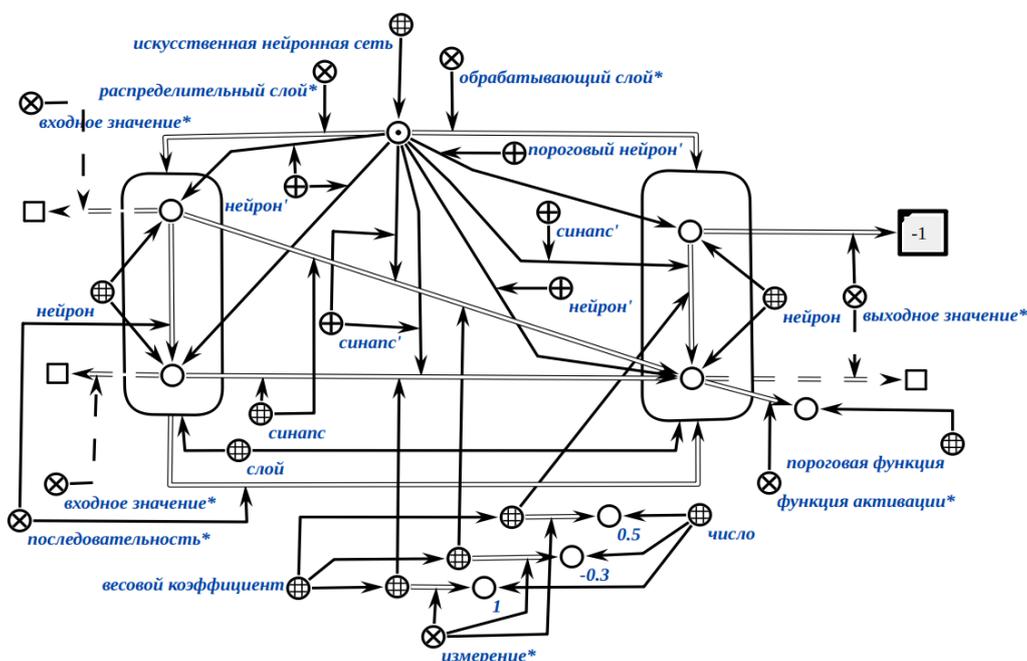


Рис. 6. Формализация в БЗ полносвязной двухслойной ИНС

Fig. 6. Formalization of the full-bonded two-layer ANN in the knowledge base

Операционная семантика языка представления ИНС задается в виде программы на некотором языке программирования, который может быть как внешним по отношению к *ostis*-системе, так и внутренним (на данный момент SCP). Каждому такому языку программирования (SCP, Python, C++ и т. д.) соответствует некоторая дочерняя предметная область Про ИНС. Следовательно, для интерпретации ИНС в БЗ операционная семантика языка представления ИНС уточняется Про ИНС на SCP.

SCP-программа – это последовательность обобщенных спецификаций (шаблонов) *scp*-операторов. Каждый *scp*-оператор является действием в памяти *ostis*-системы (*sc*-памяти). При интерпретации *scp*-программы абстрактный *sc*-агент создания *scp*-процессов порождает *scp*-процесс с учетом конкретных параметров интерпретации *scp*-программы, что на операционном уровне сводится к подстановке аргументов в обобщенные спецификации *scp*-операторов программы и генерации конкретных экземпляров этих программ (методов). Далее происходит добавление начального оператора в множество настоящих сущностей и начинается выполнение программы.

Таким образом, интерпретация *scp*-программы сводится к агентно-ориентированной обработке действий в *sc*-памяти, которыми являются *scp*-операторы. В случае интерпретации ИНС эти действия уточняют операционную семантику языка представления ИНС. Интерпретация ИНС декомпозируется на следующие действия:

- интерпретация слоя ИНС:
 - вычисление взвешенной суммы всех нейронов слоя;
 - вычисление функции активации всех нейронов слоя;
 - интерпретация сверточного слоя;
 - интерпретация пулинг-слоя;
- интерпретация нейрона ИНС.

Подробная спецификация перечисленных действий описана в работе [22]. Сами действия выполняются соответствующими им агентами, которые задают интерпретатор языка представления ИНС.

Наличие языка представления ИНС на SCP и его интерпретатора позволяет обеспечить интерпретацию ИНС в памяти *ostis*-системы. Наличие в БЗ не только экземпляров методов, но и понятий, описывающих их, создает основу для автоматизации процесса разработки ИНС. Для этого необходимо наличие среды разработки методов соответствующих классов. В случае ИНС речь идет об интеллектуальной среде разработки ИНС. Для такой среды выдвинуты следующие функциональные требования:

- генерация ИНС на основании спецификации задачи;
- автоматизация заданных пользователем этапов проектирования ИНС в соответствии с методикой;
- автоматизация процессов сравнения эффективности нейросетевых моделей;
- информационная поддержка пользователя на всех этапах проектирования ИНС;
- использование библиотеки обученных и предобученных ИНС.

В основе интеллектуальной среды разработки ИНС лежат соответствующие друг другу иерархии действий, задач и методов разработки ИНС. Построение иерархии соответствующих действий разработки ИНС сделано на основании этапов проектирования и обучения ИНС, которые в общем случае выполняют все разработчики ИНС. Данные этапы были описаны в работе [24]. На основании этих этапов выделена следующая иерархия действий по работе с ИНС:

- интерпретация ИНС;
- разработка ИНС:
 - проектирование ИНС:
 - выбор класса ИНС;
 - формирование спецификации входов и выходов ИНС;
 - обучение ИНС:
 - выбор метода оптимизации;
 - выбор минимизируемой функции ошибки;

- начальная инициализация ИНС;
- выбор гиперпараметров ИНС;
- обучение ИНС;
- оценка эффективности ИНС;
- обработка выборки:
 - поиск подходящей обучающей выборки;
 - формирование требований к обучающей выборке;
 - очистка выборки;
 - выявление содержательных признаков;
 - трансформация выборки;
 - разбиение выборки.

Все перечисленные действия специфицируются в рамках ПрО и онтологии ИНС, частными ПрО которой являются ПрО обучения и проектирования ИНС. Данные ПрО описаны в работе [22].

Под каждую интегрированную среду проектирования и обучения ИНС создается свой коллектив агентов, выполняющий перечисленные действия. Также возможна интеграция предобученных моделей, которые интегрированы по интерфейсу обучения. Все ПрО и агенты, обеспечивающие данный уровень интеграции, являются частью библиотеки многократно используемых компонентов.

Заключение. Выделенные уровни интеграции ИНС с БЗ позволяют систематизировать актуальные исследования в этой области интеграции. На основании анализа работ, соответствующих выделенным уровням, удалось сформулировать требования к моделям и средствам разработки ИНС и их интеграции с БЗ интеллектуальной системы, спецификация которых уточняется на каждом уровне интеграции.

Были описаны модель представления ИНС в БЗ; агентно-ориентированная модель разработки и интерпретации ИНС в БЗ, обеспечивающая интеграцию ИНС в БЗ на всех выделенных уровнях, а также метод классификации сущностей БЗ с помощью графовой нейронной сети.

Описанные модели и средства позволяют интегрировать в БЗ интеллектуальную систему любые обученные ИНС и использовать их для решения комплексных задач. Становятся возможными проектирование и обучение ИНС на основании как внешних данных, так и фрагментов БЗ, а также автоматизация процесса разработки ИНС в БЗ.

Продолжением настоящего исследования станет создание интеллектуальной среды разработки ИНС в БЗ и новых методов обработки фрагментов БЗ с помощью графовых нейронных сетей.

Список использованных источников

1. Attention Is All You Need [Electronic resource] / A. Vaswani [et al.]. – 2017. – Mode of access: <https://doi.org/10.48550/arXiv.1706.03762>. – Date of access: 20.06.2023.
2. Training Language Models to Follow Instructions with Human Feedback [Electronic resource] / L. Ouyang [et al.]. – 2022. – Mode of access: <https://doi.org/10.48550/arXiv.2203.02155>. – Date of access: 20.06.2023.
3. Radford, A. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks [Electronic resource] / A. Radford, L. Metz, S. Chintala. – 2016. – Mode of access: <https://doi.org/10.48550/arXiv.1511.06434>. – Date of access: 20.06.2023.
4. Kejrival, M. Knowledge graphs: A practical review of the research landscape [Electronic resource] / M. Kejrival // Applied Sciences. – 2022. – Vol. 13(4), no. 161. – Mode of access: <https://doi.org/10.3390/info13040161>. – Date of access: 20.06.2023.
5. Tsamoura, E. Neural-symbolic integration: a compositional perspective / E. Tsamoura, T. Hospedales, L. Michael // Proceedings of the AAI Conf. on Artificial Intelligence, held virtually, 2–9 Feb. 2021. – Vol. 35, no. 6. – P. 5051–5060. <https://doi.org/10.1609/aaai.v35i6.16639>
6. Deep Learning Meets Knowledge Graphs: A Comprehensive Survey [Electronic resource] / S. Yu [et al.]. – 2022. – Mode of access: <https://doi.org/10.21203/rs.3.rs-2021923/v1>. – Date of access: 20.06.2023.

7. Golenkov, V. Next-generation intelligent computer systems and technology of complex support of their life cycle / V. V. Golenkov, N. A. Gulyakina // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2022) : сб. науч. тр. / редкол.: В. В. Голенков [и др.]. – Минск : БГУИР, 2022. – Вып. 6. – P. 27–40.
8. Stokel-Walker, C. AI chatbots are coming to search engines – can you trust the results? [Electronic resource] / C. Stokel-Walker // Nature. – 2023. – Mode of access: <https://www.nature.com/articles/d41586-023-00423-4>. – Date of access: 20.06.2023.
9. Hoy, M. B. Alexa, Siri, Cortana, and More: An introduction to voice assistants / M. B. Hoy // Medical Reference Services Quarterly. – 2018. – Vol. 37, no. 1. – P. 81–88.
10. Wiley, V. Computer vision and image processing: A paper review / V. Wiley, T. Lucas // International J. of Artificial Intelligence Research. – 2018 – Vol. 2, no. 1. – P. 28–36. <https://doi.org/10.29099/ijair.v2i1.42>
11. A survey on state-of-the-art techniques for knowledge graphs construction and challenges ahead / A. Hur [et al.] // IEEE Fourth Intern. Conf. on Artificial Intelligence and Knowledge Engineering (AIKE), Laguna Hills, 1–3 Dec. 2021. – Laguna Hills, 2021. – P. 99–103. <https://doi.org/10.1109/AIKE52691.2021.00021>
12. A neural-symbolic approach to computer vision / A. Kroshchanka [et al.] // Communications in Computer and Information Science : Proceedings of the 11th Intern. Conf. "Open Semantic Technologies for Intelligent Systems" (OSTIS-2021), Minsk, 16–18 Sept. 2021. – Cham : Springer, 2022. – Vol. 1625. – P. 282–309. https://doi.org/10.1007/978-3-031-15882-7_15
13. Интеграция искусственных нейронных сетей с базами знаний / В. А. Головко [и др.] // Онтология проектирования. – 2018. – Т. 8, № 3(29). – С. 366–386. <https://doi.org/10.18287/2223-9537-2018-8-3-366-386>
14. Nielsen, F. Å. Inferring visual semantic similarity with deep learning and Wikidata: Introducing / F. Å. Nielsen L. K. Hansen // CEUR Workshop Proceedings : Proceedings of the First Workshop on "Deep Learning for Knowledge Graphs and Semantic Technologies" (DL4KGS) co-located with the 15th "Extended Semantic Web Conf." (ESWC 2018), Heraklion, 4 June 2018. – Heraklion, 2018. – Vol. 2106. – P. 56–61.
15. Modeling relational data with graph convolutional networks / M. Schlichtkrull [et al.] // Lecture Notes in Computer Science : Proceedings of the 15th Intern. Conf. "European Semantic Web Conf." (ESWC 2018), Heraklion, 3–7 June 2018. – Cham : Springer, 2018. – Vol. 10843. – P. 593–607. https://doi.org/10.1007/978-3-319-93417-4_38
16. Comet, M. Clustering knowledge graphs [Electronic resource] / M. Comet, L. Teresa // Seminar Paper. – 2018. – P. 1–12. – Mode of access: <https://doi.org/10.13140/RG.2.2.12494.08008>. – Date of access: 20.06.2023.
17. Saranya, A. A systematic review of explainable artificial intelligence models and applications: Recent developments and future trends / A. Saranya, R. Subhashini // Decision Analytics J. – 2023. – Vol. 7. – P. 100–230. <https://doi.org/10.1016/j.dajour.2023.100230>
18. Averkin, A. Explainable artificial intelligence: rules extraction from neural networks / A. Averkin, S. Yarushev // Proceedings of the 11th Intern. Conf. on Theory and Application of Soft Computing, Computing with Words and Perceptions and Artificial Intelligence (ICSCCW–2021), Antgalya, 26–27 Aug. 2021. – Antgalya, 2021. – Vol. 362. – P. 102–109. https://doi.org/10.1007/978-3-030-92127-9_17
19. Castelvechi, D. Can we open the black box of AI? / D. Castelvechi // Nature. – 2016. – Vol. 538, no. 7623. – P. 20–23.
20. Chomsky, N. The False Promise of ChatGPT / N. Chomsky, I. Roberts, J. Watumull. – Mode of access: <https://www.nytimes.com/2023/03/08/opinion/noam-chomsky-chatgpt-ai.html>. – Date of access: 20.06.2023.
21. Технология комплексной поддержки жизненного цикла семантически совместимых интеллектуальных компьютерных систем нового поколения / под общ. ред. В. В. Голенкова. – Минск : Бестпринт, 2023. – Гл. 3.6. – 1064 с.
22. Kovalev, M. Convergence and integration of artificial neural networks with knowledge bases in next-generation intelligent computer systems / M. V. Kovalev, A. A. Kroshchanka, V. A. Golovko // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2022) : сб. науч. тр. / редкол.: В. В. Голенков [и др.]. – Минск : БГУИР, 2022. – Вып. 6. – С. 173–186.
23. eXnet: an efficient approach for emotion recognition in the wild / M. N. Riaz [et al.] // Sensors. – 2020. – Vol. 20(4), no. 1087. – P. 1–17. <https://doi.org/10.3390/s20041087>
24. Integration of large language models with knowledge bases of intelligent systems / K. Bantsevich [et al.] // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2022) : сб. науч. тр. / редкол.: В. В. Голенков [и др.]. – Минск : БГУИР, 2023. – Вып. 7. – P. 213–218.
25. Xinyi, Z. Capsule graph neural network [Electronic resource] / Z. Xinyi, L. Chen // Intern. Conf. on Learning Representations. – 2019. – Mode of access: <https://openreview.net/forum?id=Byl8BnRcYm>. – Date of access: 20.06.2023.

References

1. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., ..., Polosukhin I. *Attention Is All You Need*, 2017. Available at: <https://doi.org/10.48550/arXiv.1706.03762> (accessed 17.06.2023).
2. Ouyang L., Wu J., Jiang X., Almeida D., Wainwright C. L., ..., Lowe R. J. *Training Language Models to Follow Instructions with Human Feedback*, 2022. Available at: <https://doi.org/10.48550/arXiv.2203.02155> (accessed 17.06.2023).
3. Radford A., Metz L., Chintala S. *Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks*, 2016. Available at: <https://doi.org/10.48550/arXiv.1511.06434> (accessed 17.06.2023).
4. Kejriwal M. Knowledge graphs: A practical review of the research landscape. *Applied Sciences*, 2022, vol. 13(4), no. 161. Available at: <https://doi.org/10.3390/info13040161> (accessed 17.06.2023).
5. Tsamoura E., Hospedales T., Michael L. Neural-symbolic integration: a compositional perspective. *Proceedings of the AAAI Conference on Artificial Intelligence, held virtually, 2–9 February 2021*. Vol. 35, no. 6, pp. 5051–5060. <https://doi.org/10.1609/aaai.v35i6.16639>
6. Yu S., Xu C., Bai X., Bai X., Kuncheerathodi R., ..., Xia F. *Deep Learning Meets Knowledge Graphs: A Comprehensive Survey*, 2022. Available at: <https://doi.org/10.21203/rs.3.rs-2021923/v1> (accessed 17.06.2023).
7. Golenkov V. V., Gulyakina N. A. *Next-generation intelligent computer systems and technology of complex support of their life cycle*. Otkrytye semanticheskie tehnologii proektirovaniya intellektual'nyh system : sbornik nauchnyh trudov [Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2022) : Research Papers Collection]. Ed. board: V. V. Golenkov, I. S. Azarov, V. A. Golovko, A. N. Gordey, N. A. Guliakina, ..., D. V. Shunkevich. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2022, iss. 6, pp. 27–40.
8. Stokel-Walker C. AI chatbots are coming to search engines – can you trust the results? *Nature*, 2023. Available at: <https://www.nature.com/articles/d41586-023-00423-4> (accessed 17.06.2023).
9. Hoy M. B. Alexa, Siri, Cortana, and more: An introduction to voice assistants. *Medical Reference Services Quarterly*, 2018, vol. 37, no. 1, pp. 81–88.
10. Wiley V., Lucas T. Computer vision and image processing: A paper review. *International Journal of Artificial Intelligence Research*, 2018, vol. 2, no. 1, pp. 28–36. <https://doi.org/10.29099/ijair.v2i1.42>
11. Hur A., Janjua N., Ahmed M. A survey on state-of-the-art techniques for knowledge graphs construction and challenges ahead. *IEEE Fourth International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), Laguna Hills, 1–3 December 2021*. Laguna Hills, 2021, pp. 99–103. <https://doi.org/10.1109/AIKE52691.2021.00021>
12. Kroshchanka A., Golovko V., Mikhno E., Kovalev M., ..., Zagorskij A. A neural-symbolic approach to computer vision. *Communications in Computer and Information Science : Proceedings of the 11th International Conference "Open Semantic Technologies for Intelligent Systems" (OSTIS-2021), Minsk, 16–18 September 2021*. Cham, Springer, 2022, vol. 1625, pp. 282–309. https://doi.org/10.1007/978-3-031-15882-7_15
13. Golovko V., Golenkov V., Ivashenko V., Taberko V., Ivaniuk D., ..., Kovalev M. *Integration of artificial neural networks with knowledge bases*. Ontologija proektirovaniya [Design Ontology], 2018, vol. 8, no. 3(29), pp. 366–386. <https://doi.org/10.18287/2223-9537-2018-8-3-366-386> (In Russ.).
14. Nielsen F. Å., Hansen, L. K. Inferring visual semantic similarity with deep learning and Wikidata: Introducing. *CEUR Workshop Proceedings : Proceedings of the First Workshop on "Deep Learning for Knowledge Graphs and Semantic Technologies" (DLAKGS) co-located with the 15th "Extended Semantic Web Conference" (ESWC 2018), Heraklion, 4 June 2018*. Heraklion, 2018, vol. 2106, pp. 56–61.
15. Schlichtkrull M., Kipf T. N., Bloem P., van den Berg R., Titov I., Welling M. Modeling relational data with graph convolutional networks. *Lecture Notes in Computer Science : Proceedings of the 15th International Conference "European Semantic Web Conference" (ESWC 2018), Heraklion, 3–7 June 2018*. Cham, Springer, 2018, vol. 10843, pp. 593–607. https://doi.org/10.1007/978-3-319-93417-4_38
16. Molinas C., Teresa L. Clustering knowledge graphs. *Seminar Paper*, 2018, pp. 1–12. Available at: <https://doi.org/10.13140/RG.2.2.12494.08008> (accessed 17.06.2023).
17. Saranya A., Subhashini R. A systematic review of explainable artificial intelligence models and applications: Recent developments and future trends. *Decision Analytics Journal*, 2023, vol. 7, pp. 100–230. <https://doi.org/10.1016/j.dajour.2023.100230>
18. Averkin A., Yarushev S. Explainable artificial intelligence: rules extraction from neural networks. *Proceedings of the 11th International Conference on Theory and Application of Soft Computing, Computing with Words and Perceptions and Artificial Intelligence (ICSCCW-2021), Antgalya, 26–27 August 2021*. Antgalya, 2021, vol. 362, pp. 102–109. https://doi.org/10.1007/978-3-030-92127-9_17
19. Castelveccchi D. Can we open the black box of AI? *Nature*, 2016, vol. 538, no. 7623, pp. 20–23.

20. Chomsky N., Roberts I., Watumull J. *The False Promise of ChatGPT*. Available at: <https://www.nytimes.com/2023/03/08/opinion/noam-chomsky-chatgpt-ai.html> (accessed 17.06.2023).

21. Tehnologija kompleksnoj podderzhki zhiznennogo cikla semanticheski sovместimyh intellektual'nyh komp'yuternyh sistem novogo pokolenija. *Technology for Comprehensive Life Cycle Support of Semantically Compatible, Next-generation Intelligent Computer Systems*. In V. V. Golenkov (ed.). Minsk, Bestprint, 2023, chap. 3.6, 1064 p. (In Russ.).

22. Kovalev M., Kroschchanka A., Golovko V. *Convergence and integration of artificial neural networks with knowledge bases in next-generation intelligent computer systems*. Otkrytye semanticheskie tehnologii proektirovaniya intellektual'nyh sistem : sbornik nauchnyh trudov [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2022) : Research Papers Collection*]. Ed. board: V. V. Golenkov, I. S. Azarov, V. A. Golovko, A. N. Gordey, N. A. Guliakina, ..., D. V. Shunkevich. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2022, iss. 6, pp. 173–186.

23. Riaz M., Shen Y., Sohail M., Guo M. eXnet: an efficient approach for emotion recognition in the wild. *Sensors*, 2020, Vol. 20(4), no. 1087, pp. 1–17. <https://doi.org/10.3390/s20041087>

24. Bantsevich K., Kovalev M., Tsishchanka V., Malinovskaya N., Andrushevich A. *Integration of large language models with knowledge bases of intelligent systems*. Otkrytye semanticheskie tehnologii proektirovaniya intellektual'nyh sistem : sbornik nauchnyh trudov [*Open Semantic Technologies for Designing Intelligent Systems (OSTIS-2023) : Research Papers Collection*]. Ed. board: V. V. Golenkov, I. S. Azarov, V. A. Golovko, A. N. Gordey, N. A. Guliakina, ..., D. V. Shunkevich. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radiojelektroniki, 2023, iss. 7, pp. 213–218.

25. Xinyi Z., Chen L. Capsule graph neural network. *International Conference on Learning Representations*, 2019. Available at: <https://openreview.net/forum?id=Byl8BnRcYm> (accessed 17.06.2023).

Информация об авторе

Ковалёв Михаил Владимирович, исследователь технических наук, кафедра интеллектуальных информационных технологий, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: kovalev@bsuir.by
<https://orcid.org/0000-0002-9606-9541>

Information about the author

Mikhail V. Kovalev, Researcher of Technical Sciences, the Department of Intelligent Information Technologies, Belarusian State University of Informatics and Radioelectronics.
E-mail: kovalev@bsuir.by
<https://orcid.org/0000-0002-9606-9541>