

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования – первый
проректор

_____ Хагуров Т.А.

подпись

« 02 » июня 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ФТД.02 «Гибридный ИИ: Математическое моделирование и МО»

Направление подготовки 02.03.03 Математическое обеспечение и
администрирование информационных систем

Направленность (профиль) Искусственный интеллект и аналитика данных

Форма обучения очная

Квалификация бакалавр

Краснодар 2026

Рабочая программа дисциплины «Гибридный ИИ: Математическое моделирование и МО» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем

Программу составил(и):

А.Д. Колотий, доцент КПМ, к.ф.-м.н., доцент

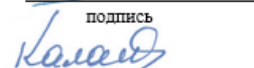
И.О. Фамилия, должность, ученая степень, ученое звание

Г.В. Калайдина, доцент, к.ф.-м.н.

И.О. Фамилия, должность, ученая степень, ученое звание



подпись

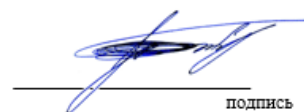


подпись

Рабочая программа дисциплины утверждена на заседании кафедры анализа данных и искусственного интеллекта

протокол № 10 «14» мая 2026 г.

Зав. кафедрой Коваленко А.В.

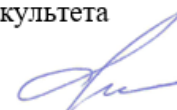


подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики

протокол № 03 «15» мая 2026 г.

Председатель УМК факультета Добровольская Н.Ю.



подпись

Рецензенты:

Мостовой Евгений Викторович, генеральный директор ООО «Портал-Юг»,

e-mail: mostovoy@portal-yug.ru

Луценко Евгений Вениаминович, доктор экономических наук, кандидат технических наук, профессор кафедры компьютерных технологий и систем Федерального государственного бюджетного образовательного учреждения высшего образования «Кубанский государственный аграрный университет имени И.Т. Трубилина», e-mail: prof.lutsenko@gmail.com

1 Цели и задачи изучения дисциплины (модуля)

1.1 Цель освоения дисциплины

Формирование у студентов системного подхода к созданию интеллектуальных систем нового поколения, которые сочетают в себе предсказательную силу машинного обучения и причинно-следственную, объяснимую природу математических моделей, основанных на физических законах и экспертных знаниях.

1.2 Задачи дисциплины

- Знакомство с историей развития интеллектуальных информационных систем, современным состоянием дисциплины и перспективами развития AGI.
- Изучение моделей представления структурированных знаний и возможностей их использования совместно с LLM.
- Изучение архитектуры экспертных систем, систем управления знаниями организации и других прикладных систем ИИ, возможностей их использования с LLM.
- Изучение основ инженерии онтологий и semantic web, технологии построения RAG-систем с использованием LLM и онтологий.
- Сформировать у студентов понимание архитектуры, методов и инструментов для построения интеллектуальных систем, помогающих лицу, принимающему решения (ЛПР), в сложных, слабоструктурированных ситуациях.

Получение практического опыта реализации экспертных систем и систем поддержки принятия решений с использованием классических методов представления структурированных знаний и LLM.

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Гибридный ИИ: Математическое моделирование и МО» относится к факультативным дисциплинам.

Требованием к «входным» знаниям (пререквизиты):

- Дифференциальные уравнения (ОДУ/УРЧП) и численные методы
- Глубокое обучение (PyTorch, backprop, автодифференцирование)
- Линейная алгебра, матанализ, Python

Постреквизиты:

- Фронтирные исследования ИИ

Смежные:

- Интеллектуальные методы оптимизации,
- Технологии обработки больших данных.

Пререквизиты: *Управление жизненным циклом ИИ-продукта (DL-3.1 (С)), Глубокое обучение (DL-3.1 (С)), Технологии компьютерного зрения (DL-3.1 (Б)); Глубокое обучение (DL-2.1 (С))*

Интеллектуальные методы оптимизации (FC-1.1(С)), Математические модели нейронных сетей (FC-1.1(Б))

Технологии интеллектуальной поддержки принятия решений (FC-2.1(С))

Постреквизиты: *Генеративный искусственный интеллект (DL-2.1 (П))*

1.4 Профессиональные роли в структуре образовательной программы

Роль 1: Data Analyst (Аналитик данных)

Задачи:

1. Статистический анализ, визуализация данных, предварительная обработка.
2. Создание прогнозных моделей
3. Построение аналитических моделей для поддержки бизнес-решений.

Роль 2: MLOps (Специалист по эксплуатации ИИ)

Задачи:

1. DevOps для ML.
2. Автоматизация, мониторинг ML-систем.
3. Операционное управление жизненным циклом ML-моделей.

Роль 3: AI PM (Менеджер проектов ИИ)

Задачи:

1. Управление ИИ-проектами от идеи до внедрения
2. Анализ бизнес-требований и постановка задач
3. Оценка эффективности и ROI ИИ-решений

1.5 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

Код и наименование индикатора	Результаты обучения по дисциплине
DL-2 Способен применять и (или) разрабатывать современные архитектуры генеративных глубоких сетей	
DL-2.1 Применяет известные архитектуры генеративных глубоких нейронных сетей для решения прикладной задачи (генерация текста, генерация изображений по тексту, синтез речи и т.д.), при необходимости проводя дообучение на наборах данных	(II) Знать архитектуры генерации (текст → изображение), мультимодальное обучение, энкодер-декодер, внимание, методы дообучения (few-shot, zero-shot), RL для генерации. Уметь модифицировать архитектуры, разрабатывать гибридные подходы (диффузия + GAN), оптимизировать под аппаратные платформы. Владеть навыками адаптации и дообучения генеративных моделей для конкретных задач и вычислительных сред.
DL-3 Способен применять и (или) разрабатывать алгоритмы, методы и технологии компьютерного зрения	
DL-3.1 Применяет (проводя выбор и эксперименты) известные алгоритмы и библиотеки компьютерного зрения, предобученные глубокие нейросетевые модели для прикладных задач анализа изображений и видеопотока, при необходимости дообучая и валидируя на собственных наборах данных	(II) Знать марковские и условные случайные поля, SIFT, трансформерные архитектуры, дообучение ViT, метрическое обучение (контрастивные, триплетные, угловые функции потерь), OCR (CRNN, внимание, трансформеры). Уметь исследовать и внедрять модели анализа изображений, выбирать функции потерь, реализовывать распознавание текста. Владеть навыками выбора технологий для систем видеоаналитики, поиска по изображениям и распознавания текста.
FC-1 Способен проводить передовые исследования в области архитектур, алгоритмов МО, оптимизации и математики	

Код и наименование индикатора	Результаты обучения по дисциплине
FC-1.1 Разрабатывает фундаментальные основы и новые алгоритмы машинного обучения	(П) Знать фундаментальные математические основы МО: теоремы аппроксимации, сходимости, гильбертовы пространства с воспроизводящим ядром, гипотезу о многообразии, DeepONet. Уметь внедрять математические идеи в обучение без сторонних библиотек, создавать новые методы и архитектуры, работать с передовыми концепциями представления знаний. Владеть навыками разработки новых алгоритмов МО на основе продвинутого математического аппарата
FC-2 Способен проводить передовые исследования в области фундаментальных и генеративных моделей	
FC-2.1 Исследует и разрабатывает большие языковые модели (LLM) и другие модели для символьных данных	(П) Знать нейросимволические парадигмы, дифференцируемое логическое программирование, логические регуляризаторы, генерацию на основе грамматик. Уметь разрабатывать новые парадигмы нейросимволической генерации, внедрять символические знания в модели. Владеть методами создания логически ограниченных генеративных моделей и нейросимволических интерфейсов.
FC-3 Способен проводить передовые исследования в области управления, решения, агентных и мультиагентных систем	
FC-3.2 Исследует и создает агентные системы	(П) Знать внутреннюю мотивацию, иерархическое RL, мета-оптимизацию, мультиагентный перенос знаний. Уметь создавать гибридные архитектуры, формулировать основы обучения в частично наблюдаемых средах, внедрять системы непрерывного самообучения агентов. Владеть навыками разработки агентных систем с непрерывным обучением и мультиагентным взаимодействием.

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 2 зач. ед. (72 часа), их распределение по видам работ представлено в таблице

Вид учебной работы	Всего часов	Семестры (часы)
		7
Контактная работа, в том числе:	52,2	52,2
50	50	50
Занятия лекционного типа	16	16
Лабораторные занятия	34	34

Занятия семинарского типа (семинары, практические занятия)			
Иная контактная работа:		2,2	2,2
Контроль самостоятельной работы (КСР)		2	2
Промежуточная аттестация (ИКР)		0,2	0,2
Самостоятельная работа, в том числе:		19,8	19,8
Проработка учебного (теоретического) материала		10	10
Выполнение индивидуальных заданий (типовой расчет)		4	4
Подготовка к текущему контролю		5,8	5,8
Контроль:		-	-
Подготовка к экзамену		-	-
Общая трудоемкость	час.	72	72
	в том числе контактная работа	52,2	52,2
	зач. ед	2	2

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.
Разделы (темы) дисциплины, изучаемые в 7 семестре

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
1.	Введение в ИИ	2	2			
2.	Представление знаний	14	4		6	4
3.	Экспертные системы	16	2		10	4
4.	Инженерия онтологий и Semantic Web	12	4		4	4
5.	Системы поддержки принятия решений	20	4		12	4
6.	Защита экзаменационных проектов	5,8			2	3,8
ИТОГО по разделам дисциплины		69,8	16		34	19,8
Контроль самостоятельной работы (КСР)		2				
Промежуточная аттестация (ИКР)		0,2				
Подготовка к текущему контролю						
Общая трудоемкость по дисциплине		72				

Примечание: Л – лекции, ПЗ – практические занятия/семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

2.3 Содержание разделов (тем) дисциплины

2.3.1 Занятия лекционного типа

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
1	Введение в гибридный ИИ и интеллектуальные системы	История, философия и определения ИИ. Классификация интеллектуальных систем. Эволюция от символического ИИ к машинному обучению и большим языковым моделям (LLM). Понятие гибридного ИИ. Введение в искусственный общий интеллект (AGI) и этику.	Р

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
2	Классические методы представления знаний и решения задач	Продукционные системы и стратегии логического вывода. Методы поиска решений (А*, анализ средств и целей). Формализмы представления знаний: логика, семантические сети, фреймы. Базы знаний. Введение в обучение с подкреплением.	ЛР
3	Современные подходы к представлению знаний и планированию	Автоматическое планирование (STRIPS), составление расписаний. Нечеткая логика и сети Петри для работы с неопределенностью и параллельными процессами. Интеграция LLM для генерации, интерпретации и объяснения планов.	ЛР
4	Экспертные и адаптивные системы	Архитектура экспертных систем, механизмы вывода и обработки неопределенности. Эволюция towards адаптивных, динамических и персонализированных систем (на примере рекомендательных систем). Использование LLM для улучшения интерфейсов и объяснений в экспертных системах.	ЛР
5	Инженерия онтологий и Semantic Web	Онтологии как основа для представления знаний. Методологии и инструменты построения (Protege). Технологический стек Semantic Web: RDF, RDFS, OWL. Логический вывод. Запросы к онтологиям с помощью SPARQL.	ЛР
6	Гибридные системы: интеграция онтологий и LLM	Практическое применение онтологий (управление знаниями, семантический поиск). Синергия онтологий и LLM: RAG-архитектура для снижения галлюцинаций, обоснование ответов, автоматизация обогащения онтологий. Критический анализ возможностей и ограничений.	ЛР
7	Системы поддержки принятия решений (СППР): от данных к решениям	Архитектура и классификация СППР. Подсистема управления данными (ETL, OLAP, Data Mining). Количественные методы (математическое моделирование, MCDA) и качественные методы (экспертные системы, нечеткая логика) в подсистеме управления моделями.	ЛР
8	Современные СППР: машинное обучение, LLM-агенты и кейсы	Применение машинного обучения (прогнозная аналитика, рекомендательные системы) и ХАИ в СППР. Современные тенденции: когнитивные системы, аналитика в реальном времени, дашборды. Разработка и этика LLM-агентов для поддержки принятия решений. Разбор сквозных кейсов.	ЛР

Примечание: ЛР – отчет/защита лабораторной работы, КП - выполнение курсового проекта, КР - курсовой работы, РГЗ - расчетно-графического задания, Р - написание реферата, Э - эссе, К - коллоквиум, Т – тестирование, РЗ – решение задач.

2.3.2 Занятия семинарского типа (практические / семинарские занятия/ лабораторные работы)

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
1.	Представление знаний	Фреймы Реализация иерархии фреймов для представления знаний.	Т, ЛР
2.		Семантические сети. Построение и обход семантической сети	Т, ЛР
3.		Нечеткая логика. Построение системы нечеткого вывода.	Т, ЛР

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
4.	Экспертные системы	Экспертные системы. Разработка простой экспертной системы для диагностики	Т, ЛР
5.		Поиск в пространстве состояний. Решение задачи с использованием алгоритма A*	Т, ЛР
6.		Планирование. Автоматическое планирование на основе STRIPS-модели	Т, ЛР
7.		Обработка естественного языка (ЕЯ). Создание простого символьного чат-бота с обработкой ЕЯ	Т, ЛР
8.		Составление расписаний. Решение задачи оптимизации расписания	Т, ЛР
9.	Инженерия онтологий и Semantic Web	Protege. Создание онтологии в Protege	Т, ЛР
10.		OWL + LLM. Интеграция онтологий с большими языковыми моделями	
11.	Системы поддержки принятия решений	Инструменты и среды разработки. Jupyter Notebook, библиотеками Python для анализа данных (Pandas, NumPy, Matplotlib). Загрузка и первичный анализ набора данных, построение базовых графиков.	Т, ЛР
12.		Работа с данными и OLAP. Задача: Создание простого ETL-пайплайна и анализ с помощью Pandas. Загрузка данных из CSV, их очистка и агрегация (аналог операций OLAP: slice, dice, roll-up, drill-down).	Т, ЛР
13.		Реализация методов многокритериального решения. Реализация метода анализа иерархий (АНР). Написание кода на Python для попарного сравнения критериев выбора и расчета приоритетов.	Т, ЛР
14.		Построение экспертной системы с нечеткой логикой. Задача: Использование библиотеки scikit-fuzzy. Создание нечеткой системы для управления уровнем сервиса в ресторане на основе факторов "качество еды" и "качество обслуживания".	Т, ЛР
15.		Разработка прогнозной модели для СПИР. Задача: Построение и оценка модели машинного обучения. Использование scikit-learn для создания модели классификации (дерево решений или случайный лес) на реальном наборе данных. Визуализация дерева решений.	Т, ЛР
16.		Создание интерактивного дашборда. Визуализация результатов для ЛПР. Использование библиотеки Plotly Dash или Streamlit для создания простого веб-дашборда, отображающего ключевые метрики и прогнозы из предыдущих практик.	Т, ЛР
17.		Защита проектов.	ЛР

Примечание: ЛР – отчет/защита лабораторной работы, КП – выполнение курсового проекта, КР – курсовой работы, РГЗ – расчетно-графического задания, Р – написание реферата, Э – эссе, К – коллоквиум, Т – тестирование, РЗ – решение задач.

2.3.4 Примерная тематика курсовых работ (проектов)

Не предусмотрены

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Изучение теоретического материала	Методические указания по организации самостоятельной работы студентов, утвержденные УМК ФКТиПМ, протокол №1 от 28.08.2025
2	Решение задач	Методические указания по организации самостоятельной работы студентов, утвержденные УМК ФКТиПМ, протокол №1 от 28.08.2025

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,
- в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии

В соответствии с требованиями ФГОС в программа дисциплины предусматривает использование в учебном процессе следующих образовательные технологии: чтение лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

- Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.

- Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.

- Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.

- Информационно-коммуникационные технологии (ИКТ) - расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:

- Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.

– Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.

– Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.

– Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.

– Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.

– Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.

– Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеперечисленных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Основные виды интерактивных образовательных технологий включают в себя:

– работа в малых группах (команде) - совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;

– проектная технология - индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;

– анализ конкретных ситуаций - анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;

– развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

Подход разбора конкретных задач и ситуаций широко используется как преподавателем, так и студентами во время лекций, лабораторных занятий и анализа результатов самостоятельной работы. Это обусловлено тем, что при исследовании и решении каждой конкретной задачи имеется, как правило, несколько методов, а это требует разбора и оценки целой совокупности конкретных ситуаций.

Темы, задания и вопросы для самостоятельной работы призваны сформировать навыки поиска информации, умения самостоятельно расширять и углублять знания, полученные в ходе лекционных и практических занятий.

Подход разбора конкретных ситуаций широко используется как преподавателем, так и студентами при проведении анализа результатов самостоятельной работы.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4. Оценочные и методические материалы

4.1 Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «название дисциплины».

Оценочные средства включает контрольные материалы для проведения **текущего контроля** в форме тестовых заданий, кейсов и **промежуточной аттестации** в форме вопросов и заданий к **экзамену**.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине (модулю) предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

– в печатной форме увеличенным шрифтом,

– в форме электронного документа.

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

Структура оценочных средств для текущей и промежуточной аттестации

№ п/п	Контролируемые разделы (темы) дисциплины*	Код контролируемой компетенции (или ее части)	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	Введение в гибридный ИИ и интеллектуальные системы	FC-2.1	ЛР №1, 2, 4	Вопросы: 1, 2 Рефераты: 1, 2, 20

2	Классические методы представления знаний и решения задач	FC-2.1, FC-1.1	ЛР №1, 2, 3, 5	Вопросы: 3, 4, 5 Рефераты: 2, 3, 4
3	Современные подходы к представлению знаний и планированию	FC-1.1, FC-3.2, DL-2.1	ЛР №3, 6, 7, 8	Вопросы: 14, 18 Рефераты: 5, 16
4	Экспертные и адаптивные системы	FC-2.1, DL-2.1	ЛР №4, 7, 14	Вопросы: 6, 13 Рефераты: 1, 11, 19
5	Инженерия онтологий и Semantic Web	FC-2.1	ЛР №9, 10	Вопросы: 7, 8, 9, 10 Рефераты: 3, 6, 7, 8
6	Гибридные системы: интеграция онтологий и LLM	FC-2.1, DL-2.1	ЛР №10	Вопросы: 11, 12 Рефераты: 6, 7, 8, 9, 10
7	Системы поддержки принятия решений (СППР): от данных к решениям	FC-1.1, FC-2.1, DL-3.1	ЛР №11, 12, 13, 14	Вопросы: 15, 16 Рефераты: 11, 12, 13
8	Современные СППР: машинное обучение, LLM-агенты и кейсы	FC-3.2, DL-3.1, DL-2.1	ЛР №15, 16, 17	Вопросы: 17, 18, 19 Рефераты: 14, 15, 17, 18

Показатели, критерии и шкала оценки сформированных компетенций

№ п/п	Код и наименование индикатора	Результаты обучения	Наименование оценочного средства	
			Текущий контроль	Промежу- точная аттестация
Соответствие освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: удовлетворительно /зачтено)				
на пороговом уровне:				
1	DL-2.1 Применяет известные архитектуры генеративных глубоких сетей	- Умеет работать с LLM через API - Формирует базовые промпты - Понимает принципы RAG-архитектуры - Различает типы генеративных моделей	ЛР №7, 10	Вопросы: 11, 12, 13 Рефераты: 6, 9, 10
2	DL-3.1 Применяет алгоритмы и библиотеки компьютерного зрения	- Выполняет базовый EDA данных - Работает с библиотеками pandas, numpy - Строит простые визуализации - Понимает pipeline обработки данных	ЛР №11, 12, 15	Вопросы: 15, 17 Рефераты: 12, 13
3	FC-1.1 Разрабатывает фундаментальные основы и новые алгоритмы МО	- Понимает математические основы методов - Реализует стандартные алгоритмы - Работает с библиотеками scikit-learn, scikit-fuzzy - Анализирует результаты моделей	ЛР №3, 5, 13, 14, 15	Вопросы: 4, 14, 16, 17 Рефераты: 3, 12, 14
4	FC-2.1 Исследует и разрабатывает LLM и модели для символьных данных	- Создает простые онтологии в Protégé - Формирует SPARQL-запросы - Разрабатывает базы правил для экспертных систем - Понимает принципы символьного вывода	ЛР №1, 2, 4, 9, 10, 14	Вопросы: 2, 5, 6, 7, 8, 9, 10 Рефераты: 1, 2, 4, 7, 8, 11
5	FC-3.2 Исследует и создает агентные системы	- Разрабатывает компоненты СППР - Создает интерактивные дашборды - Понимает принципы планирования - Реализует базовые агентные взаимодействия	ЛР №6, 8, 16, 17	Вопросы: 18, 19 Рефераты: 5, 15, 17, 18

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Практические задания по тематике лабораторных работ

1. Фреймы

Представить в виде системы фреймов и метафреймов, записать в EAV-таблицу, записать в виде программы Prolog или Python:

а) Катя, София и Мария -- женщины, Сергей, Петр, Иван и Даниил -- мужчины. Иван родитель Кати, Сергей родитель Софии и Даниила, Катя родитель Даниила, Даниил родитель Марии и Петра.

Описать предикаты:

Пример: Сын,
woman(katrin). Дочь,
woman(sofia). Мама,
woman(mary). Папа,
man(sergey). Бабушка,
man(daniel). Дедушка,
man(petr). Дядя,
man(ivan). Тетя,
parent(ivan,katrin). Предок ancestor(A, B),
parent(sergey,sofia). Родственник relative(A, B)
parent(sergey,daniel).
parent(katrin,daniel).
parent(daniel,mary).
parent(daniel,petr).

Записать последовательность унификаций, подстановок, резолюций и откатов при вычислении цели: relative(petr, sofia).

б) Краснодар - столица Краснодарского края. Население Краснодара 1 млн. Краснодарский край граничит с Ростовской областью и Ставропольским краем. Столица Ростовской области - город Ростов-на-Дону с населением 1.5 млн. В Ростовской области находится город Таганрог с населением 250 тыс. Река Кубань протекает по Краснодарскому краю и Ростовской области.

в) Студент университета обучается в группе. На каждом курсе несколько групп. Бакалавр пишет диссертацию у научного руководителя на 4 курсе. Некоторые бакалавры поступают в магистратуру. Магистр пишет диссертацию у научного руководителя на 2м году обучения.

2. Семантические сети

а) Нарисуйте семантическую сеть для задачи 1 а).

б) Нарисуйте семантическую сеть для задачи 1 б).

в) Нарисуйте семантическую сеть для задачи 1 в).

г) Нарисуйте семантическую сеть для базы знаний:

```
cond(1, 'кормит детенышей молоком').  
cond(2, 'имеет перья').  
cond(3, 'плавает').  
cond(4, 'ест мясо').  
cond(5, 'имеет копыта').  
cond(6, 'летает').  
cond(7, 'откладывает яйца').  
cond(8, 'имеет шерсть').  
cond(9, 'имеет полосы').  
cond(10, 'имеет пятна').  
cond(11, 'имеет черно-белую окраску').  
rule('гепард', [1,4,8,10]).  
rule('тигр', [1,4,8,9]).  
rule('зебра', [1,5,8,9,11]).  
rule('пингвин', [2,3,11]).  
rule('орел', [2,6]).  
rule('кит', [1,3,11]).
```

д) Представьте следующие предложения в виде семантической сети и фактов в прологе:

Иван дает своим студентам много трудных заданий. Мария дает студентам Ивана огромное количество сложных заданий.

Предположим, нам нужно написать систему, которая определит кто дает студентам Ивана больше всех заданий. Как можно это сделать?

3. Экспертные системы

Реализовать экспертную систему на Prolog, должна задавать вопросы пользователю и находить подходящий ответ:

- от 10 фактов,
- от 5 правил,
- возможность кроме да и нет отвечать неопределенно,
- возможность более чем одного варианта ответа,
- обучение (при отсутствии подходящего ответа или не правильном ответе должна спрашивать правильный ответ и пополнять базу знаний дописывая факты в файл).

4. Нечеткая логика

а) Почитать спецификацию FCL (IEC 1331 part 7, файл [iec_1131_7_cd1.pdf](#)).

б) Почитать описание, примеры и API библиотеки jFuzzyLogic (каталог Docs и сайт <http://jfuzzylogic.sourceforge.net/>). Разобрать и запустить TestTipper.java и документацию библиотеки simplful

в) Придумать нечеткий контроллер какого-либо механизма, например климат-контроль, насос для откачки воды, автономный робот-марсоход, беспилотный летательный аппарат. Контроллер должен содержать минимум четыре входные переменные, две выходные, по три правила на каждую выходную переменную. Задать контроллер в виде FCL-файла. Продемонстрировать его работу на разных значениях входных переменных с помощью программы на Java или Python. С помощью LLM реализовать интерфейс к нечеткому контроллеру на естественном языке.

5. Поиск в пространстве состояний. По шаблону поиска в пространстве состояний написать программу (реализовать 'start', 'goal', 'next_state' и 'safe_state' на Prolog) для поиска минимального решения одной из задач:

Фермер-волк-коза-капуста
(классика)

Миссионеры и каннибалы
3 миссионера и 3 каннибала на одной стороне, перевести на другую живыми, в лодке от одного до 2-х человек. Если каннибалы оказываются в большинстве, то они съедают миссионера.

Ревнивые мужья
Три ревнивых мужа переправляются с женами через реку в двухместной лодке. Как переправить, чтобы ни одна жена с чужим мужем не оставалась в лодке или на берегу.

6. Поиск A*

Реализовать "пятнашки" алгоритмом A*, реализовать проверку на заикливание, составить эвристическую функцию для цели:

1 2 3 4
5 6 7 8
9 10 11 12
13 14 15

7. Проект ИИ (АВ-поиск, обучение с подкреплением)

Реализовать антагонистическую игру (человек против компьютера) с использованием поиска в пространстве состояний на глубину два или более ходов, использовать алгоритм АВ-отсечения и обучение с подкреплением для функции игры. Одна игра на трех человек, игры не должны повторяться в группе. Примеры хороших игр: рендзю (сложно), шашки (сложно), уголки, реверси, 3D крестики-нолики, крестики нолики по 4 с гравитацией (просто). Компьютер должен выигрывать у преподавателя!

8. Планирование в мире блоков

Блоки лежат на столе или друг на друге. Блоки не могут висеть в воздухе. На один блок можно поставить не более одного блока. Например:

on(a, table).

on(c, a).

on(b, table).

За одно действие можно переставить только один блок. Написать планировщик действий по достижению определенного расположения блоков из заданного начального, используя подход анализа целей и средств.

Реализовать защиту целей, см. пример

plan([on(b, c), on(a, b), on(c, table)]).

Сформулировать в предикатах понятие правильного списка целей в зависимости от того, может ли данный список целей быть полностью выполнен для мира физических блоков.

Например, [on(a, b), on(b, a)] — некорректно. Написать предикат, который бы анализировал корректность перемещений блоков. Подсказка:

- стол не может быть сверху чего-либо
- нельзя двигать несуществующие блоки
- блок не может быть сверху себя
- нельзя поставить два блока на один
- нельзя поставить один блок на два блока
- все блоки должны на чем-то лежать
- ...

Переписать на списках без assert и retract, явно хранить список удаления и добавления (не обязательное задание).

9. Обработка ЕЯ

а) С помощью DCG построить контекстно-зависимую грамматику на русском языке.

б) Идиоматический интерфейс в мире блоков

Для мира блоков сформулировать идиомы «what is block X sitting on?», «which blocks are on the table?», «put all blocks in single pile.», «put the block on top of X on top of block Y (or on the table).» С помощью LLM классифицировать произвольную команду на ЕЯ как одну из идиом с необходимыми параметрами и вызвать планировщик.

в) Идиоматический интерфейс экспертной системы

Для экспертной системы 3 сформулировать не менее 5 идиом вида «Какие животные умеют летать?», «Кто кормит детенышей молоком?» и т.д. Строки из базы знаний по возможности не должны дублироваться в грамматике. С помощью LLM классифицировать произвольную команду на ЕЯ как одну из идиом экспертной системы и вызвать соответствующую идиому.

10. Составление расписаний, программирование в ограничениях

Реализовать задачу 8 ферзей через CLP-FD. Разобрать программу составления расписаний школьных занятий Маркуса Триски. Прочитать его диссертацию про CLP-FD в SWI-Prolog. С помощью LLM преобразовать пожелания преподавателя на естественном языке в CLP-FD ограничение в системе расписаний.

11. Онтология

С помощью Protege построить онтологию для задач 1 а) и б). Построить онтологию по индивидуальному заданию, предметные области не должны совпадать в группе, продемонстрировать на примере все свойства отношений (рефлексивное, симметричное, транзитивное, функциональное, инверсно-функциональное и т.д.).

12. Разработка прототипа СППР для задачи индустриального партнера

Цель: Применить полученные знания для создания комплексного прототипа системы, интегрирующей данные, модели и удобный интерфейс.

Этапы выполнения проекта:

Выбор предметной области и формулировка задачи: (Пример: "СППР для выбора инвестиционного портфеля", "СППР для диагностики заболеваний растений", "СППР для управления запасами на складе", выбрать из кейсов, предоставленных индустриальными партнерами).

Анализ требований и проектирование архитектуры: Описать компоненты (данные, модели, интерфейс).

Реализация:

Подсистема данных: сбор и предобработка данных.

Подсистема моделей: реализация одной или нескольких моделей (например, MCDA для выбора + ML для прогноза).

Подсистема диалога: создание простого веб-интерфейса (например, на Streamlit).

Тестирование и валидация: Проверка работы системы на тестовых данных, оценка качества прогнозов.

Документирование и презентация:

Пояснительная записка с описанием архитектуры, алгоритмов и результатов.

Презентация с демонстрацией работы прототипа.

Код проекта в репозитории (GitHub/GitLab).

Критерии оценки:

Актуальность и сложность задачи.

Корректность применения методов ИИ.

Качество программной реализации и интерфейса.

Полнота и ясность документации.

Умение презентовать и защитить свой проект.

Для студентов, участвующих в соревнованиях по робототехнике, данная задача может быть заменена на защиту проекта по управлению роботом с применением машинного зрения (распознавание объектов) или обучения с подкреплением.

Пример лабораторной работы

Лабораторная работа №2: Представление знаний

Тема: Построение семантической сети и фреймовой модели

Цель работы:

Научиться представлять знания в виде семантической сети и фреймовой модели, а также применять эти методы для моделирования предметной области.

Задачи:

Построить семантическую сеть для предметной области (например, "Смартфоны").

Реализовать фреймовую модель на основе этой сети.

Описать отношения IS-A и Part-Of.

Создать EV-таблицу для одного из фреймов.

Проанализировать эффективность и удобство обоих подходов.

Ожидаемые результаты:

Семантическая сеть в графическом виде.

Фреймовая модель в табличном или текстовом виде.

EV-таблица для одного из объектов.

Отчет с анализом применения методов представления знаний.

Ход работы

1. Выбор предметной области

Выбрать область, например:

- "Смартфоны"
- "Автомобили"
- "Птицы"
- "Игровые ноутбуки"

2. Построение семантической сети

Пример:

- Узлы: iPhone 15, смартфон, Apple, камера, экран, процессор
- Связи: iPhone 15 IS-A смартфон, iPhone 15 HAS камера, камера HAS разрешение, процессор MANUFACTURED_BY Apple.

Рекомендуется использовать инструменты:

- draw.io
- yEd
- Graphviz
- рукописная схема (в отчете можно отсканировать)

3. Построение фреймовой модели

Пример фрейма:

(Frame: iPhone 15
(IS-A: смартфон)
(HAS: камера)
(HAS: экран)
(HAS: процессор)
(MANUFACTURED_BY: Apple)
)

Каждый объект должен быть представлен как отдельный фрейм. Описать слоты и их значения.

4. Создание EV-таблицы

Для одного из объектов (например, камера iPhone 15) создать EV-таблицу:

СЛОТ	ЗНАЧЕНИЕ	ЕДИНИЦА ИЗМЕРЕНИЯ
Разрешение	12 мегапикселей	Мп
Апертура	f/1.8	-
Оптическая стабилизация	Да	-

5. Анализ эффективности

- Какие отношения легче выражаются в семантической сети?
- Какие преимущества дают фреймы?
- Какие сложности возникли при моделировании?
- В каких ИИ-системах может использоваться эта модель?

Требования к отчету

1. Титульный лист

Название работы, ФИО студента, группа, дата.

2. Введение

Цель и задачи работы.
Описание датасета (объем, типы ошибок).

3. Теоретическая часть

Кратко описать:

- Что такое семантические сети
- Что такое фреймы
- Что такое EV-таблицы
- Отношения IS-A и Part-Of

4. Реализация

Схема семантической сети (можно вставить как изображение)

Текстовое представление фреймов

EV-таблица

Комментарии к выбору слотов и связей.

5. Результаты

Выводы по применимости методов.

Сравнение наглядности и удобства семантической сети и фреймов.

6. Выводы

Проблемы, возникшие при обработке.

Предложения по улучшению.

7. Приложения

Исходные файлы (схемы, фреймы, таблицы)

Словарь терминов (если использовались специфичные).

Критерии оценки

Отлично (5): Полная реализация, грамотная модель, четкие выводы, высокая наглядность.

Хорошо (4): Хорошая реализация, но есть мелкие недочеты в модели или выводах.

Удовлетворительно (3): Выполнены основные этапы, но модель не полностью соответствует предметной области.

Неудовлетворительно (2): Не выполнены ключевые этапы, модель не отражает смысл.

Рекомендуемые инструменты

- draw.io / yEd / Graphviz – для построения семантической сети
- Текстовый редактор (Notepad++, Word) – для оформления фреймов и EV-таблиц
- Python или Prolog – для представления фреймов в памяти программы.

Тематика рефератов (с фокусом на гибридный ИИ)

Блок 1: Символьный ИИ и его интеграция с современными подходами

1. Нейро-символьный искусственный интеллект (NeSy): Современное состояние и перспективы.
2. Эволюция экспертных систем: От MYCIN к гибридным системам с LLM.
3. Нечеткие онтологии: Представление и рассуждение с нечеткими знаниями в Semantic Web.

4. Базы знаний против больших языковых моделей: Сравнительный анализ как оснований для интеллектуальных систем.
5. Автоматическое планирование в гибридных средах: Роль LLM в генерации и исполнении планов.

Блок 2: Онтологии и семантические технологии в эпоху LLM

6. RAG-архитектура на основе онтологий: Повышение достоверности ответов больших языковых моделей.
7. Обучение онтологий (Ontology Learning) с использованием LLM: Автоматизация извлечения знаний из текста.
8. Семантические Reasoning в гибридных системах: Как логический вывод обогащает генеративные модели.
9. Промпт-инжиниринг для взаимодействия с онтологиями и базами знаний.
10. Графы знаний (Knowledge Graphs) как основа для корпоративных гибридных ИИ-систем.

Блок 3: Гибридные системы поддержки принятия решений

11. Гибридные СППР для медицинской диагностики: Объединение символьных правил, прогнозных моделей ML и LLM.
12. Использование нечеткой логики и машинного обучения для управления рисками в финансовых СППР.
13. Разработка гибридной рекомендательной системы: Символьная фильтрация по онтологии и коллаборативная фильтрация.
14. Многоагентные LLM-системы (на примере AutoGen) для решения сложных бизнес-задач.
15. Этические аспекты и проблема доверия к гибридным системам поддержки принятия решений.

Блок 4: Прикладные аспекты и кейсы

16. Гибридный ИИ в робототехнике: Интеграция символьного планирования, компьютерного зрения и LLM для навигации и взаимодействия.
17. Применение гибридного ИИ для интеллектуального анализа научной литературы и ускорения исследований.
18. Кейс: Построение гибридного ИИ-помощника для технической поддержки на основе RAG и онтологии продукта.
19. Гибридные интеллектуальные обучающие системы (ИОС): Персонализация обучения с помощью символьной модели ученика и генеративного контента.
20. Будущее гибридного ИИ: На пути к искусственному общему интеллекту (AGI).

Тестовые вопросы по дисциплине

1. Введение в системы ИИ

Какое утверждение наиболее точно описывает ИИ?

- a) Программа, которая может играть в шахматы
- b) Система, способная воспринимать среду, учиться и действовать рационально
- c) Автомат, выполняющий жёстко заданные инструкции
- d) Система, способная выполнять любые задачи быстрее человека

2. Экспертные системы

Что не является компонентом экспертной системы?

- a) Машина вывода
- b) База знаний
- c) Генератор отчетов
- d) Интерфейс пользователя

3. Продукционные системы

Какой метод вывода начинается с целей и ищет пути их достижения?

- a) Прямой
- b) Обратный
- c) Смешанный
- d) Гибридный

4. Prolog

Какой механизм в Prolog позволяет вернуться к предыдущему выбору при неудаче?

- a) Отсечение
- b) Возврат (backtracking)
- c) Унификация
- d) Рекурсия

5. Фреймы

Как называется элемент фрейма, описывающий его характеристику?

- a) Атрибут
- b) Слот
- c) Значение
- d) Дескриптор

6. Семантические сети

Какой из подходов к машинному переводу не является семантическим?

- a) Статистический
- b) Синтаксический
- c) Логический
- d) Глубокий семантический

7. Обработка естественного языка

Какой инструмент в Prolog используется для описания грамматики?

- a) DCG
- b) CFG
- c) NLP
- d) DTD

8. Поиск в пространстве состояний

Какой алгоритм использует эвристическую функцию для поиска кратчайшего пути?

- a) BFS
- b) DFS
- c) A*
- d) Dijkstra

9. Нейронные сети

Какой тип нейронной сети лучше всего подходит для распознавания изображений?

- a) Рекуррентная
- b) Полносвязная
- c) Сверточная
- d) Радиально-базисная

10. Планирование действий

Какой метод планирования использует декомпозицию целей?

- a) A*

- b) BFS
- c) GPS
- d) STRIPS

11. Адаптивные системы

Какой метод используется для персонализации контента в обучающих системах?

- a) Кластеризация
- b) Рекомендательные системы
- c) Линейная регрессия
- d) PCA

12. RDF и графовые модели

Какой формат данных используется для описания информации в виде триплетов?

- a) JSON
- b) XML
- c) RDF
- d) CSV

13. Semantic Web

Какой язык используется для описания онтологий в Semantic Web?

- a) RDF
- b) RDFS
- c) OWL
- d) XML

14. Нечеткая логика

Какое множество описывается с помощью функции принадлежности?

- a) Четкое
- b) Нечеткое
- c) Случайное
- d) Дискретное

15. Интеллектуальные агенты

Какой тип агента наиболее близок к человеку?

- a) Реактивный
- b) Адаптивный
- c) Рациональный
- d) Простой

Зачетно-экзаменационные материалы для промежуточной аттестации (зачет)

Контрольные вопросы к зачету по дисциплине «Гибридный ИИ: Математическое моделирование и МО»

Раздел 1: Фундаментальные концепции и представление знаний

1. Дайте определение гибридного искусственного интеллекта. Назовите причины его востребованности и приведите пример гибридной системы.
2. В чем заключаются ключевые различия между символьным и субсимвольным (связонистским) подходами в ИИ? Перечислите их сильные и слабые стороны.
3. Опишите архитектуру продукционной системы. В чем разница между прямой и обратной цепочкой рассуждений и для каких задач они оптимальны?

4. Что такое эвристическая функция в алгоритме A*? Каким условиям она должна удовлетворять, чтобы гарантировать оптимальность решения?
5. Сравните фреймы и семантические сети как модели представления знаний. Какой из подходов более пригоден для представления иерархических понятий и почему?

Раздел 2: Экспертные системы и онтологии

6. Назовите основные компоненты экспертной системы и их функции. Какую роль играет подсистема объяснений?
7. Что такое онтология в контексте информатики? Опишите ее основные компоненты (классы, экземпляры, свойства, аксиомы).
8. Каково назначение языков RDFS и OWL? Почему RDFS недостаточно для построения выразительных онтологий, и какие новые возможности предоставляет OWL?
9. Как механизм логического вывода (reasoner) использует онтологию для проверки непротиворечивости и автоматической классификации?
10. Для чего используется язык запросов SPARQL? Приведите пример типичного запроса к семантическим данным.

Раздел 3: Интеграция методов: Гибридный ИИ и LLM

11. Что такое архитектура RAG (Retrieval-Augmented Generation) и как она использует синергию онтологий и больших языковых моделей?
12. Опишите, как онтологии помогают преодолеть ключевые ограничения LLM, такие как галлюцинации и отсутствие актуальных знаний.
13. Каковы потенциальные роли LLM в жизненном цикле экспертной системы (от проектирования до эксплуатации)?
14. Как нечеткая логика позволяет работать с неопределенностью в гибридных системах поддержки принятия решений? Приведите пример.

Раздел 4: Системы поддержки принятия решений (СППР) и машинное обучение

15. Опишите архитектуру системы поддержки принятия решений (СППР). Как взаимодействуют ее основные подсистемы?
16. Какие методы многокритериального принятия решений (например, метод анализа иерархий - АНР) вы знаете? В чем их суть?
17. Как машинное обучение интегрируется в СППР? Почему интерпретируемость моделей (XAI) критически важна в этом контексте?
18. Что такое LLM-агент? Из каких компонентов он состоит и как обеспечивается надежность его действий (например, с помощью критикагентов или планирования)?
19. Назовите и охарактеризуйте современные тенденции в развитии СППР (когнитивные системы, аналитика в реальном времени, гибридные агенты).

Практические задания к зачету

1. Напишите базу знаний на языке Prolog или Python для модели родственных отношений.
Реализуйте предикаты родитель/2, муж/1, жен/1, и определите на их основе предикаты отец/2, мать/2, брат/2, сестра/2.
2. Реализуйте простую продукционную систему с механизмом прямого вывода. Создайте 5 правил и рабочую память. Продемонстрируйте, как система выбирает и применяет правила.
3. Постройте семантическую сеть для предметной области "Автомобили". Укажите узлы и связи (IS-A, HAS, PART-OF). Опишите графически или в текстовом виде.
4. Создайте фреймовую модель на основе семантической сети из задания 3. Представьте фреймы в виде структур с использованием слотов и значений.
5. Рассчитайте вероятность наступления события с помощью наивного Байесовского классификатора.

- Используйте простой набор данных (например: прогноз погоды — будет ли дождь?).
6. Напишите код для поиска кратчайшего пути с помощью алгоритма A*. *
Используйте граф с 6–8 узлами и заданными весами рёбер и эвристикой.
7. Создайте онтологию для предметной области "Птицы" с использованием RDF/OWL (в текстовом виде).
Определите не менее 3 классов, 2 свойств и 2 подклассов.
8. Примените нечеткую логику для оценки "качества обслуживания" в ресторане.
Определите лингвистические переменные ("низкое", "среднее", "высокое"), функции принадлежности и правило вывода.
9. Опишите простого рационального агента для задачи "поиск выхода из лабиринта".
Опишите его перцепты, возможные действия и функцию рациональности.

Вопросы на анализ и сравнение

1. Сравните продукционные и фреймовые системы: особенности, сильные и слабые стороны, сферы применения.
2. Почему онтологии предпочтительнее словарей и таксономий при построении интеллектуальных систем?
3. Какие преимущества и недостатки у Prolog по сравнению с императивными языками при реализации ИИ-систем?
4. Чем отличается представление знаний в семантических сетях и в логических системах? Какой подход более удобен для машинного вывода?

4.2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Методические рекомендации, определяющие процедуры оценивания на зачете:

Процедура промежуточной аттестации проходит в соответствии с Положением о текущем контроле и промежуточной аттестации обучающихся ФГБОУ ВО «КубГУ».

Итоговой формой контроля сформированности компетенций у обучающихся по дисциплине является экзамен. Студенты обязаны сдать экзамен в соответствии с расписанием и учебным планом.

ФОС промежуточной аттестации состоит из вопросов к экзамену и результатов текущего контроля.

Экзамен по дисциплине преследует цель оценить работу студента за курс, получение теоретических знаний, их прочность, развитие творческого мышления, приобретение навыков самостоятельной работы, умение применять полученные знания для решения практических задач.

Форма проведения экзамена: устно.

Экзаменатору предоставляется право задавать студентам дополнительные вопросы по всей учебной программе дисциплины.

Результат сдачи экзамена заносится преподавателем в экзаменационную ведомость и зачетную книжку.

Оценивание уровня освоения дисциплины основывается на качестве выполнения студентом заданий текущего контроля и ответов на вопросы экзамена.

Критерии оценки:

1. Оценка ответов на вопросы зачета (30% итоговой оценки)

Отлично (5), зачтено

Полные, развернутые ответы с демонстрацией глубокого понимания темы.

Использование примеров, формул, корректных терминов.

Умение анализировать и сравнивать методы (например, производственные системы vs. фреймы).

% выполнения: 90–100% (допускаются незначительные неточности).

Хорошо (4), зачтено

Ответы содержат основные идеи, но без углубленного анализа.

Возможны небольшие ошибки в деталях или формулировках.

% выполнения: 75–89%.

Удовлетворительно (3), зачтено

Ответы поверхностные, с существенными пробелами.

Отсутствие примеров или некорректное применение терминов.

% выполнения: 60–74%.

Неудовлетворительно (2), не зачтено

Отсутствие понимания ключевых концепций.

Грубые ошибки или неспособность ответить на большую часть вопросов.

% выполнения: <60%.

2. Оценка выполнения лабораторных работ (60% итоговой оценки)

Отлично (5)

Полное выполнение всех этапов проектов и лабораторных работ, включая интеграцию LLM. Создание антагонистической игры с элементами обучения с подкреплением. Создание СППР.

Четкая документация кода и анализ результатов.

% выполнения: 90–100%.

Хорошо (4)

Выполнены основные задачи лабораторных работ, но в проектах имеются существенные недочеты. Интеграция LLM не выполнена или работает в большинстве случаев некорректно.

% выполнения: 75–89%.

Удовлетворительно (3)

Решены базовые задачи лабораторных работ, проекты выполнены с критическими ошибками.

Низкое качество кода или отсутствие анализа.

% выполнения: 60–74%.

Неудовлетворительно (2)

Невыполнение ключевых этапов.

Код нерабочий или отсутствует.

% выполнения: <60%.

3. Оценка тестовых вопросов (10% итоговой оценки)

Отлично (5)

13–15 правильных ответов (87–100%).

Демонстрация уверенного владения терминологией и методами.

Хорошо (4)

10–12 правильных ответов (67–80%).

Незначительные ошибки.

Удовлетворительно (3)

7–9 правильных ответов (47–60%).

Путаница в базовых концепциях.

Неудовлетворительно (2)

Менее 7 правильных ответов (<47%).

Неспособность отличить ключевые модели представления знаний.

Итоговая оценка (суммарно)

Оценка	Экзамен (30%)	Практика (60%)	Тест (10%)	Общий %
Отлично (5)	90–100%	90–100%	87–100%	≥85%
Хорошо (4)	75–89%	75–89%	67–80%	70–84%
Удовлетворительно (3)	60–74%	60–74%	47–60%	55–69%
Неудовлетворительно (2)	<60%	<60%	<47%	<55%

Для допуска к экзамену необходимо выполнить **все лабораторные работы на минимум "удовлетворительно"**.

"Отлично" требует высоких результатов во всех компонентах (особенно в практических заданиях).

Практические кейсы оцениваются по:

- Корректности кода.
- Достижению метрик.
- Качеству отчета (анализ ошибок, визуализация).

Методические рекомендации, определяющие процедуры оценивания лабораторных работ:

Процедура оценивания лабораторных работ проходит в соответствии с Положением о текущем контроле и промежуточной аттестации обучающихся ФГБОУ ВО «КубГУ».

По каждой лабораторной работе оформляется отчет. Отчеты сдаются на проверку руководителю в течение курса по мере их выполнения, и защищаются студентами в установленном порядке.

При защите отчета студенту могут быть заданы вопросы и дополнительные задания по сути лабораторной работы, в том числе из списка контрольных вопросов к данной лабораторной работе. При неудовлетворительной оценке знаний студента по теме данного отчета, студент возвращается к повторному изучению соответствующих материалов, после чего допускается к повторной защите. Неудовлетворительно выполненный отчет также возвращается на доработку.

Отчет должен содержать заголовок, тему лабораторной работы, цель, задание, индивидуальную тему, описание хода выполнения работы, необходимые прикладные материалы (схемы, макеты документов и т.п.), в соответствии с требованиями к содержанию, и выводы по работе.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

- при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;
- при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;
- при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,

– в форме электронного документа.

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4.3 Методические указания по организации лабораторных работ по дисциплине "Гибридный ИИ: Математическое моделирование и МО"

1. Общие положения

1.1. Цель лабораторного практикума

Формирование у студентов практических навыков разработки гибридных интеллектуальных систем, сочетающих символьные методы представления знаний и современные нейросетевые подходы.

1.2. Задачи лабораторного практикума

– Освоение технологий представления знаний (фреймы, семантические сети, онтологии)

– Приобретение навыков работы с экспертными системами и системами поддержки принятия решений

– Освоение методов интеграции символьного ИИ с большими языковыми моделями

– Разработка комплексных гибридных ИИ-систем

2. Программное обеспечение и технические требования

2.1. Базовое программное обеспечение

Категория	Программное обеспечение	Назначение
Язык программирования	Python 3.8+	Основная среда разработки
Среда разработки	Jupyter Notebook, VS Code с расширениями Python	Выполнение и документирование работ
Система контроля версий	Git, GitHub/GitLab	Хранение и совместная работа над кодом
Контейнеризация	Docker, Docker Compose	Создание воспроизводимых сред выполнения

2.2. Специализированные библиотеки Python

Для символьного ИИ и представления знаний:

python

Экспертные системы и логический вывод

experta==1.9.0 *# Продукционные системы*

pyknow==0.3.0 *# Другой фреймворк для экспертных систем*

rdflib==6.3.2 *# Работа с RDF, SPARQL*

owlready2==0.39 *# Работа с OWL онтологиями*

Математические модели и оптимизация

scikit-fuzzy==0.4.2 *# Нечеткая логика*

python-constraint==1.4.0 # Задачи с ограничениями
ortools==9.6.2534 # Оптимизация от Google

Графы и сети

networkx==3.1 # Анализ графов, семантические сети

Для работы с данными и ML:

python

Обработка данных

pandas==2.0.3

numpy==1.24.3

scipy==1.10.1

Машинное обучение

scikit-learn==1.3.0

shap==0.42.1 # Объяснимый ИИ (XAI)

Визуализация

matplotlib==3.7.1

seaborn==0.12.2

plotly==5.14.1

Для интеграции с LLM и современным ИИ:

python

Работа с LLM

openai==0.28.1 # OpenAI API

langchain==0.0.346 # Цепи вызовов LLM

llama-index==0.8.42 # RAG-архитектуры

transformers==4.33.3 # Локальные модели

Дашборды и интерфейсы

streamlit==1.27.0 # Веб-интерфейсы для СППР

gradio==3.44.0 # Интерактивные интерфейсы

2.3. Специализированное ПО

- **Protégé Desktop** 5.5.0+ - визуальное редактирование онтологий
- **GraphDB Free** 10.0+ - семантическая база данных
- **Apache Jena Fuseki** 4.0+ - SPARQL-сервер

3. Организация лабораторных работ

3.1. Подготовительный этап

1. **Установка ПО:** Единоразовая установка базового окружения
2. **Клонирование репозитория:** Получение шаблонов и учебных материалов
3. **Создание виртуального окружения:** Изоляция зависимостей проекта

3.2. Структура отчета по лабораторной работе

Каждый отчет должен содержать:

1. Титульный лист
 2. Цель и задачи работы
 3. Теоретическую справку (кратко)
 4. Листинг кода с комментариями
 5. Результаты выполнения (скриншоты, графики)
 6. Выводы и анализ полученных результатов
 7. Ответы на контрольные вопросы
4. Методические рекомендации по выполнению работ

4.1. Модуль 1: Представление знаний

ЛР 1-3: Фреймы, семантические сети, нечеткая логика

Рекомендации:

- Начинать с простых предметных областей (библиотека, университет)
- Использовать объектно-ориентированный подход для фреймов
- Визуализировать семантические сети с помощью networkx
- Экспериментировать с функциями принадлежности в нечеткой логике

Типичные ошибки:

- Слишком сложная предметная область на начальном этапе
- Отсутствие визуализации промежуточных результатов
- Неправильная настройка функций принадлежности

4.2. Модуль 2: Экспертные системы**ЛР 4-8: Экспертные системы, поиск, планирование****Рекомендации:**

- Использовать библиотеку experta для быстрого прототипирования
- Тестировать системы на различных входных данных
- Сравнивать эффективность разных алгоритмов поиска
- Документировать базу правил в отдельном файле

Контрольные точки:

- Работоспособность механизма вывода
- Качество генерируемых объяснений
- Эффективность работы алгоритмов поиска

4.3. Модуль 3: Онтологии и Semantic Web**ЛР 9-10: Protégé, OWL + LLM****Рекомендации:**

- Освоить базовые конструкции OWL перед работой с Protégé
- Проверять онтологии на непротиворечивость
- Использовать готовые онтологии для обучения (FOAF, Dublin Core)
- Тестировать SPARQL-запросы на различных endpoint'ax

Интеграция с LLM:

- Начинать с простых RAG-архитектур
- Сравнивать качество ответов с онтологией и без
- Анализировать случаи галлюцинаций LLM

4.4. Модуль 4: Системы поддержки принятия решений**ЛР 11-16: Сквозной проект СППР****Рекомендации:**

- Выбрать реалистичную предметную область
- Поэтапно развивать функциональность системы
- Использовать Streamlit для быстрого создания интерфейса
- Тестировать систему на различных сценариях принятия решений

Критерии оценки:

- Полнота реализации заявленной функциональности
- Качество визуализации и интерфейса
- Обоснованность принимаемых решений
- Качество кода и документации

5. Оценка результатов**5.1. Критерии оценки лабораторных работ**

- **Отлично:** Полная реализация всех задач, качественный код, глубокий анализ
- **Хорошо:** Реализация основных задач, незначительные недочеты
- **Удовлетворительно:** Реализация базовой функциональности, существенные недочеты
- **Неудовлетворительно:** Работа не выполнена или выполнена неверно

5.2. Требования к защите работ

1. Демонстрация работоспособности системы
 2. Ответы на вопросы по реализации
 3. Анализ возникших проблем и путей их решения
 4. Предложения по улучшению системы
6. Рекомендации для преподавателей

6.1. Организация рабочего процесса

- Проводить вводный инструктаж по каждой лабораторной работе
- Обеспечить студентов шаблонами кода и примерами
- Использовать систему контроля версий для отслеживания прогресса
- Проводить промежуточные консультации

6.2. Дифференциация заданий

- **Базовый уровень:** Реализация по готовому алгоритму
- **Продвинутый уровень:** Оптимизация и улучшение производительности
- **Творческий уровень:** Разработка собственной предметной области

7. Ресурсы и литература

7.1. Основные ресурсы

- Официальная документация библиотек Python
- Онлайн-курсы по специализированным технологиям
- Научные статьи по гибриднему ИИ

7.2. Рекомендуемая литература

1. "Artificial Intelligence: A Modern Approach" - Russell, Norvig
2. "Knowledge Representation and Reasoning" - Brachman, Levesque
3. Документация по OWL и Semantic Web (W3C)
4. Научные публикации по нейро-символьному ИИ

8. Техническая поддержка

Для решения технических проблем предусмотрены:

- Форум курса для обсуждения общих вопросов
- Индивидуальные консультации по техническим проблемам
- База знаний с решениями типичных проблем
- Готовые Docker-образы с настроенным окружением

Данные методические рекомендации обеспечивают системный подход к выполнению лабораторных работ и формированию у студентов практических навыков в области гибридного искусственного интеллекта.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

5.1 Учебная литература:

1. Рассел, Стюарт. Искусственный интеллект : современный подход. Т. 2. Знания и рассуждения в условиях неопределенности / С. Рассел, П. Норвиг ; перевод с английского и редакция А. В. Слепцова. - 4-е изд. - Москва ; Санкт-Петербург : Диалектика, 2021. - 475 с. : ил. - ISBN 978-5-907365-26-1. - ISBN 978-5-907365-24-7. - ISBN 978-0-13-461099-3 : 3600 р. - Текст : непосредственный. URL: http://212.192.134.46/MegaPro/UserEntry?Action=Link_FindDoc&id=255397&idb=0
2. Бессмертный, И. А. Системы искусственного интеллекта : учебник для вузов / И. А. Бессмертный. - 3-е изд., испр. и доп. - Москва : Юрайт, 2025. - 164 с. - URL: <https://urait.ru/bcode/561602> (дата обращения: 16.04.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-534-18416-7. - Текст : электронный. URL: http://212.192.134.46/MegaPro/UserEntry?Action=Link_FindDoc&id=147094&idb=0

3. Бессмертный, И. А. Интеллектуальные системы : учебник и практикум для вузов / И. А. Бессмертный, А. Б. Нугуманова, А. В. Платонов. - 2-е изд. - Москва : Юрайт, 2025. - 250 с. - URL: <https://urait.ru/bcode/558664> (дата обращения: 16.04.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-534-20734-7. - Текст : электронный. URL: http://212.192.134.46/MegaPro/UserEntry?Action=Link_FindDoc&id=144873&idb=0
4. Воронов, М. В. Системы искусственного интеллекта : учебник и практикум для вузов / М. В. Воронов, В. И. Пименов, И. А. Небаев. - 2-е изд., перераб. и доп. - Москва : Юрайт, 2023. - 268 с. - URL: <https://urait.ru/bcode/532212> (дата обращения: 14.11.2023). - Режим доступа: для авториз. пользователей. - ISBN 978-5-534-17032-0. - Текст : электронный. URL: http://212.192.134.46/MegaPro/UserEntry?Action=Link_FindDoc&id=271957&idb=0
5. Sun, X., Li, J., Kovalenko, A.V., Feng, W., Ou, Y. Integrating Reinforcement Learning and Learning From Demonstrations to Learn Nonprehensile Manipulation //IEEE Transactions on Automation Science and Engineering, 2023, 20(3), 1735–1744, DOI: 10.1109/TASE.2022.3185071, Q1
6. Petukhova, A.V.; Kovalenko, A.V.; Ovsyannikova, A.V. Algorithm for Optimization of Inverse Problem Modeling in Fuzzy Cognitive Maps. Mathematics 2022, 10, 3452. DOI: 10.3390/math10193452, Q1
7. Kirillova, E.; Kovalenko, A.; Urtenov, M. Study of the Current–Voltage Characteristics of Membrane Systems Using Neural Networks. AppliedMath 2025, 5, 10. <https://doi.org/10.3390/appliedmath5010010>
8. Kadurin, Artur, et al. "The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology." Oncotarget 8.7 (2016): 10883.
9. Kadurin, Artur, et al. "druGAN: an advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico." Molecular pharmaceutics 14.9 (2017): 3098-3104.
10. Polykovskiy, Daniil, et al. "Molecular sets (MOSES): a benchmarking platform for molecular generation models." Frontiers in pharmacology 11 (2020): 565644.
11. Khrabrov, Kuzma, et al. "\$\nabla^2\$ DFT: A Universal Quantum Chemistry Dataset of Drug-Like Molecules and a Benchmark for Neural Network Potentials." Advances in Neural Information Processing Systems 37 (2024): 36869-36889.
12. Polykovskiy, Daniil, et al. "Entangled conditional adversarial autoencoder for de novo drug discovery." Molecular pharmaceutics 15.10 (2018): 4398-4405.
13. Николенко, Сергей, Кадури, Артур и Архангельская Екатерина. Глубокое обучение. Издательский дом "Питер", 2017..

5.2. Периодические издания и конференции (А*):

1. IEEE Transactions on Big Data – научные статьи по обработке больших данных.
2. Journal of Big Data (SpringerOpen) – открытый журнал с исследованиями в области Big Data.
3. Big Data Research (Elsevier) – публикации по анализу, управлению и визуализации данных.
4. Data Science Journal (CODATA) – междисциплинарные исследования данных.
5. ACM Transactions on Knowledge Discovery from Data (TKDD) – методы извлечения знаний из больших данных.
6. <https://openreview.net/forum?id=FMMF1a9ifL>
7. <https://openreview.net/forum?id=ElUrNM9U8c#discussion>
8. <https://openreview.net/forum?id=JoO6mtCLHD>
9. <https://aclanthology.org/2024.findings-emnlp.760/>
10. <https://aclanthology.org/2020.coling-main.588/>
11. https://link.springer.com/chapter/10.1007/978-3-030-72113-8_30
12. https://link.springer.com/chapter/10.1007/978-3-031-42448-9_10

13. <https://aclanthology.org/2024.findings-naacl.288/>

5.3. Интернет-ресурсы, в том числе современные профессиональные базы данных и информационные справочные системы

Электронно-библиотечные системы (ЭБС):

1. ЭБС «ЮРАЙТ» <https://urait.ru/>
2. ЭБС «УНИВЕРСИТЕТСКАЯ БИБЛИОТЕКА ОНЛАЙН» <http://www.biblioclub.ru/>
3. ЭБС «BOOK.ru» <https://www.book.ru>
4. ЭБС «ZNANIUM.COM» www.znanium.com
5. ЭБС «ЛАНЬ» <https://e.lanbook.com>

Профессиональные базы данных

1. Scopus <http://www.scopus.com/>
2. ScienceDirect <https://www.sciencedirect.com/>
3. Журналы издательства Wiley <https://onlinelibrary.wiley.com/>
4. Научная электронная библиотека (НЭБ) <http://www.elibrary.ru/>
5. Полнотекстовые архивы ведущих западных научных журналов на Российской платформе научных журналов НЭИКОН <http://archive.neicon.ru>
6. Национальная электронная библиотека (доступ к Электронной библиотеке диссертаций Российской государственной библиотеки (РГБ) <https://rusneb.ru/>
7. "Лекториум ТВ" <http://www.lektorium.tv/>
8. Университетская информационная система РОССИЯ <http://uisrussia.msu.ru>

Информационные справочные системы

1. Консультант Плюс - справочная правовая система (доступ по локальной сети с компьютеров библиотеки)

Ресурсы свободного доступа

1. КиберЛенинка <http://cyberleninka.ru/>;
2. Американская патентная база данных <http://www.uspto.gov/patft/>
3. Министерство науки и высшего образования Российской Федерации <https://www.minobrnauki.gov.ru/>;
4. Федеральный портал "Российское образование" <http://www.edu.ru/>;
5. Информационная система "Единое окно доступа к образовательным ресурсам" <http://window.edu.ru/>;
6. Единая коллекция цифровых образовательных ресурсов <http://school-collection.edu.ru/> .
7. Проект Государственного института русского языка имени А.С. Пушкина "Образование на русском" <https://pushkininstitute.ru/>;
8. Справочно-информационный портал "Русский язык" <http://gramota.ru/>;
9. Служба тематических толковых словарей <http://www.glossary.ru/>;
10. Словари и энциклопедии <http://dic.academic.ru/>;
11. Образовательный портал "Учеба" <http://www.ucheba.com/>;
12. Законопроект "Об образовании в Российской Федерации". Вопросы и ответы http://xn--273--84d1f.xn--p1ai/voprosy_i_otvety

Собственные электронные образовательные и информационные ресурсы КубГУ

1. Электронный каталог Научной библиотеки КубГУ <http://megapro.kubsu.ru/MegaPro/Web>
2. Электронная библиотека трудов ученых КубГУ <http://megapro.kubsu.ru/MegaPro/UserEntry?Action=ToDb&idb=6>
3. Среда модульного динамического обучения <http://moodle.kubsu.ru>

4. База учебных планов, учебно-методических комплексов, публикаций и конференций <http://infoneeds.kubsu.ru/>
5. Библиотека информационных ресурсов кафедры информационных образовательных технологий <http://mschool.kubsu.ru/>
6. Электронный архив документов КубГУ <http://docspace.kubsu.ru/>
7. Электронные образовательные ресурсы кафедры информационных систем и технологий в образовании КубГУ и научно-методического журнала "ШКОЛЬНЫЕ ГОДЫ" <http://icdau.kubsu.ru/>

6. Методические указания для обучающихся по освоению дисциплины (модуля)

По курсу предусмотрено проведение лекционных занятий, на которых дается систематизированный материал по основным направлениям искусственного интеллекта. В ходе лекций рассматриваются ключевые концепции: от представления знаний (семантические сети, фреймы, онтологии) до современных подходов, таких как большие языковые модели, интеллектуальные агенты и системы управления знаниями. Особое внимание уделяется практическим аспектам: разбираются примеры применения ИИ в реальных задачах (экспертные системы, автоматическое планирование), обсуждаются типичные проблемы и ограничения при внедрении ИИ-решений. После каждой лекции рекомендуется выполнение практических заданий для закрепления ключевых понятий и методов.

Лабораторные занятия курса посвящены практическому освоению методов построения систем искусственного интеллекта. На занятиях студенты реализуют и анализируют модели представления знаний, работают с продукционными системами, фреймами и семантическими сетями, а также исследуют работу алгоритмов поиска (A*, BFS, DFS), работают с нечеткой логикой. Используются реальные сценарии, такие как построение онтологий, моделирование агентов, реализация простых экспертных систем. Студенты применяют такие языки, как Prolog, RDF/OWL, Java, Python. Используют знания, полученные в других курсах, для интеграции больших языковых моделей с ИИ на базе структурированных знаний. После каждого лабораторного занятия предлагаются задания для самостоятельного выполнения — от построения семантических сетей до применения алгоритмов вывода.

При самостоятельной работе студентам необходимо изучать рекомендованную литературу (учебники, научные статьи, документацию библиотек) для глубокого понимания теоретических основ и современных подходов в области искусственного интеллекта. Выполняя проектные задания, студент должен уметь: формулировать задачу (например, моделирование знаний, поиск решения, построение агента); подбирать и адаптировать подходящие методы (от классических логических систем до вероятностных и нейросетевых подходов); реализовывать и оценивать эффективность выбранных решений. Особое внимание уделяется навыкам анализа и сравнения методов, а также критическому осмыслению их применимости в конкретной предметной области.

Важнейшим компонентом курса является самостоятельная проектная работа, в ходе которой студент разрабатывает законченное решение на основе ИИ-технологий — например, антагонистическую игру, интеллектуального агента, систему на основе нечеткой логики. Такой проект позволяет закрепить навыки проектирования, реализации и оценки систем искусственного интеллекта, а также научиться интегрировать разные методы и подходы в единое решение.

Для студентов с ограниченными возможностями здоровья предусмотрены дополнительные индивидуальные консультации, на которых преподаватель подробно разъясняет сложные аспекты дисциплины, помогает адаптировать практические задания и обеспечивает специальные условия для освоения методов работы с системами

искусственного интеллекта. Индивидуальный подход позволяет таким студентам полноценно участвовать в учебном процессе и достигать требуемых результатов обучения.

А. Применение результатов дисциплины «Гибридный ИИ: Математическое моделирование и МО» в кейсах ПАО «Сбербанк»

Кейс 1: Разработка гибридной системы для скоринга МСБ

Задача:

Создать гибридную систему скоринга для кредитования малого и среднего бизнеса (МСБ), сочетающую данные из различных источников, экспертные правила регулятора и прогнозные модели ML.

Проблема:

Традиционные скоринговые модели на основе ML плохо интерпретируются и не учитывают экспертные знания и законодательные ограничения ЦБ РФ. Необходимо повысить точность прогноза дефолта и обеспечить прозрачность принятия решений для внутреннего контроля и проверок регулятора.

Техническое задание:

1. **Разработать символьную базу знаний:** Реализовать онтологию предметной области "Кредитование МСБ", включающую понятия: Компания, Владелец, Отрасль, КредитнаяИстория, ФинансовыеПоказатели, Залог. Закодировать в онтологии жесткие регуляторные правила (например, "если отрасль в 'красном' списке ЦБ, то кредит не выдавать").

2. **Интеграция с LLM:** Реализовать RAG-модуль на основе онтологии. LLM должна использоваться для анализа текстовых отчетов бизнеса и извлечения структурированной информации (например, определение стабильности доходов из описания деятельности).

3. **Гибридная модель принятия решений:** Построить pipeline, в котором: Сначала применяются жесткие символьные правила из онтологии. Затем данные передаются в ML-модель (например, градиентный бустинг) для оценки вероятности дефолта.

На основе выходов ML-модели и извлеченных LLM фактов система формирует итоговое решение (Одобрить/Отклонить) и **развернутое обоснование** на естественном языке.

Требуемые технологии: OWLReady2 / Protégé (онтология), LangChain (RAG, работа с LLM), Scikit-learn / CatBoost (ML-модель), Experta (экспертные правила).

Проверяемые компетенции: FC-2.1, DL-2.1, FC-1.1, FC-3.2.

Кейс 2: Интеллектуальный консультант по инвестиционным продуктам

Задача:

Разработать гибридного ИИ-агента для персонализированной консультации клиентов частного банковского обслуживания по инвестиционным продуктам.

Проблема:

Существующие системы часто предлагают шаблонные решения, не учитывающие глубоко профиль риска клиента, его финансовые цели и текущую рыночную ситуацию.

Необходим адаптивный помощник, который ведет осмысленный диалог, опирается на актуальные данные и объясняет свои рекомендации.

Техническое задание:

1. **Построение профиля клиента:** Создать динамическую онтологию "Инвестор", которая обновляется на основе диалога, истории операций и загруженных документов (анализ LLM). Онтология должна включать УровеньРиска, ФинансовыеЦели, ГоризонтИнвестирования, Предпочтения.

2. **Многоагентная архитектура:** Реализовать систему из нескольких LLM-агентов:

Агент-интервьюер: Ведет диалог с клиентом, уточняет цели.

Агент-аналитик: Анализирует текущие рыночные данные (через API), новости и фундаментальные показатели активов.

Агент-критик: Проверяет предлагаемые агентом-аналитиком портфели на соответствие онтологическим правилам (диверсификация, соответствие риску).

3. **Генерация персонализированного предложения:** Агенты совместно формируют инвестиционную рекомендацию и подробное обоснование в виде связного текста и визуализации (дашборд в Streamlit).

Требуемые технологии: AutoGen / CrewAI (многоагентные системы), OpenAI API / GPT, Streamlit (дашборд), RDFLib (онтология).

Проверяемые компетенции: FC-3.2, DL-2.1, FC-2.1.

Кейс 3: Оптимизация логистики и управления наличностью в банковских отделениях

Задача:

Создать гибридную систему поддержки принятия решений (СППР) для прогнозирования потребности в наличных деньгах и планирования инкассации для сети отделений Сбербанка.

Проблема:

Неоптимальные графики инкассации приводят к избытку "замороженной" наличности в одних отделениях и ее нехватке в других. Необходимо учесть dozens факторов: исторические операции, дни недели, праздники, местные события, погоду, транспортную логистику.

Техническое задание:

1. **Подсистема прогнозирования:** Разработать ансамбль ML-моделей (регрессия, временные ряды) для прогноза оборота наличных по каждому отделению на несколько дней вперед. Использовать Prophet и Scikit-learn.

2. **Подсистема планирования (символьный ИИ):**

– Реализовать модель планирования на основе ограничений (Constraint Satisfaction Problem). Ограничения: грузоподъемность инкассаторских машин, лимиты работы отделений, временные окна.

– Использовать нечеткую логику для учета "мягких" факторов, таких как "рискованность района" или "вероятность проведения массового мероприятия" (на основе анализа новостей LLM).

3. **Гибридный планировщик:** Интегрировать модули. Прогнозы от ML-моделей и факторы от LLM передаются в символьный планировщик, который строит оптимальный маршрут и график инкассации, минимизирующий затраты и риски.

4. **Визуализация:** Реализовать дашборд для диспетчера, отображающий рекомендованные маршруты на карте, графики загрузки и обоснование решений.

Требуемые технологии: Scikit-learn, Facebook Prophet, python-constraint / OR-Tools (планирование), scikit-fuzzy (нечеткая логика), Plotly / Streamlit (дашборд).

Проверяемые компетенции: FC-1.1, FC-3.2, DL-3.1, DL-2.1.

Общие принципы оценки кейсов:

1. **Качество гибридной архитектуры:** Насколько удачно объединены символьные и субсимвольные компоненты.

2. **Работоспособность прототипа:** Фактическая выполнимость технического задания и демонстрация работающего кода.

3. **Обоснованность решений:** Способность команды аргументировать выбор тех или иных методов и технологий.

4. **Инновационность:** Применение оригинальных подходов к решению бизнес-задач Сбера.

5. **Презентация:** Четкость и структурированность изложения решения. Эти кейсы позволяют студентам применить весь спектр полученных знаний в условиях, максимально приближенных к реальным задачам ведущей ИТ-компании страны.

Б. Применение результатов дисциплины «Гибридный ИИ: Математическое моделирование и МО» в кейсах для компании AVA Group

Кейс 1: Интеллектуальный планировщик строительных ресурсов

Задача: Создать гибридную систему для прогнозирования потребности в строительных материалах и оптимизации логистики для объектов AVA LAB.

Проблема: Срыв сроков поставок и простои на стройплощадках из-за ручного планирования и неучтенных факторов (погода, пробки, изменения в проекте).

Техническое задание:

1. **Прогнозная модель (Машинное обучение):** Реализовать модель временных рядов (например, с использованием Prophet или ARIMA), которая на основе данных о ходе выполнения предыдущих проектов, графика работ и погодных условий прогнозирует ежедневную потребность в ключевых материалах (бетон, арматура, кирпич).

2. **Экспертная система (Символьный ИИ):** Разработать в Protégé онтологию предметной области "Строительный проект", включающую понятия Объект, Стадия Строительства, Материал, Поставщик, Логистический Маршрут. Закодировать в онтологии правила, например: "Если Стадия Строительства = 'Заливка Фундамента', то необходим Материал = 'Бетон' с Приоритетом = 'Высокий'".

3. **Гибридный планировщик:** Создать модуль на Python, который объединяет прогнозы ML-модели с правилами из онтологии. Система должна автоматически формировать оптимальные заказы поставщикам и строить маршруты доставки с учетом дорожной ситуации (используя API карт).

Проверяемые компетенции: FC-1.1, FC-2.1, FC-3.2.

Кейс 2: Гибридный анализатор рисков для застройщика

Задача: Разработать систему поддержки принятия решений (СППР) для оценки рисков при запуске новых строительных проектов.

Проблема: Решения о запуске проектов часто принимаются на основе ограниченного количества финансовых показателей, без учета множества слабоструктурированных данных (новостной фон, отзывы о локации, активность в соцсетях).

Техническое задание:

1. **Извлечение информации с помощью LLM:** Настроить RAG-конвейер с использованием LangChain. LLM должна анализировать новостные статьи и отзывы о районах планируемой застройки, чтобы выявлять позитивные и негативные факторы (например, строительство новой школы или, наоборот, экологические проблемы).

2. **Символьная база знаний:** Построить онтологию "Риски Застройщика" с классами Риск, Фактор, Проект. Связать извлеченные LLM факторы с типами рисков (например, "Репутационный", "Логистический", "Экологический") в онтологии.

3. **Многокритериальная оценка:** Реализовать метод анализа иерархий (АНР) для количественной оценки проектов. Критерии для сравнения (стоимость земли, транспортная доступность, репутационные риски) и их веса должны динамически определяться на основе данных из онтологии и выводов ML-моделей.

4. **Интерактивный дашборд:** Визуализировать итоговые рейтинги проектов и детальное обоснование решений с помощью Streamlit.

Проверяемые компетенции: DL-2.1, FC-2.1, FC-1.1.

Кейс 3: AI-консьерж для новых жилых комплексов

Задача: Создать интеллектуального многоагентного помощника для новых жильцов, который помогает с заселением и адаптацией в новом районе.

Проблема: После покупки жилья клиенты сталкиваются с множеством вопросов (подключение услуг, ремонт, знакомство с инфраструктурой), что создает высокую нагрузку на службу клиентского сервиса застройщика.

Техническое задание:

1. **Многоагентная система:** Реализовать архитектуру из нескольких LLM-агентов (например, с помощью AutoGen):

Агент-юрист: Отвечает на вопросы о документах, правах и обязанностях собственника.

Агент-ремонтник: Помогает с планировкой, рекомендует проверенные бригады и отслеживает статус ремонта.

Агент-сосед: Знакомит с инфраструктурой района, рекомендует кружки для детей, магазины и т.д.

2. **Динамическая онтология жилого комплекса:** Создать и постоянно обновлять семантическую базу знаний о жилом комплексе, включающую Дом, Квартира, Жилец, Услуга, Подрядчик.

3. **Голосовой и текстовый интерфейс:** Настроить канал взаимодействия с жильцами через популярные мессенджеры или умные колонки, используя технологии распознавания и синтеза речи.

Проверяемые компетенции: FC-3.2, DL-2.1, FC-2.1.

7. Материально-техническое обеспечение по дисциплине (модулю))

Перечень информационно-коммуникационных технологий

1. Облачные платформы и сервисы

AWS/GCP/Azure/YandexCloud/cloud.ru – облачные вычисления для использования LLM

2. Системы управления версиями и коллаборации

Git/GitHub/GitLab – контроль версий кода и совместная разработка

3. Система управления обучением

Moodle – сдача работ

Перечень лицензионного и свободно распространяемого программного обеспечения

1. Свободное ПО (Open Source)

Инструменты для разработки ИИ-систем:

SWI Prolog – реализация логических и экспертных систем

Java – язык программирования общего назначения с мощной экосистемой, активно используемый в разработке ИИ-систем, особенно в корпоративной среде и образовании

Python – основной язык программирования для ИИ-приложений

Библиотеки для работы с знаниями и логикой:

networkx – построение и анализ семантических сетей

rdflib – работа с RDF и онтологиями

OWLready2 – загрузка и модификация онтологий в формате OWL

Системы управления знаниями:

Protégé – инструмент для создания онтологий

Apache Jena – платформа для работы с RDF, RDFS и OWL

Neo4j – графовая СУБД для представления знаний в виде графов

Инструменты для нечеткой логики и агентов:

jFuzzyLogic – библиотека нечеткой логики на Java

scikit-fuzzy – библиотека нечеткой логики на Python

simplful – реализация нечетких систем на Python
 Визуализация и анализ:
 Graphviz – построение графов и диаграмм (семантические сети, деревья вывода)
 Matplotlib / Plotly – визуализация результатов работы ИИ-моделей
 Streamlit / Gradio – создание веб-интерфейсов для экспертных систем и систем поддержки принятия решений
 СУБД и хранение знаний:
 SQLite / PostgreSQL – хранение структурированных данных и знаний
 MongoDB – хранение неструктурированных данных и графов знаний
 Neo4J – обработка RDF-данных

Наименование специальных помещений	Оснащенность специальных помещений	Перечень лицензионного программного обеспечения
Учебные аудитории для проведения занятий лекционного типа	Мебель: учебная мебель Технические средства обучения: экран, проектор, компьютер ауд. 129, 131, А-305, А-307	MS Office Word 2016 и выше Ms Power Point 2016 и выше
Учебные аудитории для проведения текущего контроля (Ауд. 101, 102, 105/1, 106 и 106а)	Мебель: учебная мебель Технические средства обучения: Экран, компьютер Оборудование: компьютерная техника с подключением к информационно-коммуникационной сети «Интернет» и доступом в электронную информационно-образовательную среду образовательной организации	Браузер Google Chrome, Jupyter Notebook 6.3.0 и выше (язык Python с библиотеками Numpy, Pandas, gensim, NLTK, PyMorphy, фреймворком PyTorch)
Учебные аудитории для проведения промежуточной аттестации (Ауд. 129, 131, А-305, А-307)	Мебель: учебная мебель	-
Учебные аудитории для проведения лабораторных работ (Ауд. 101, 102, 105/1, 106 и 106а)	Мебель: учебная мебель Технические средства обучения: экран, компьютер Оборудование: компьютерная техника с подключением к информационно-коммуникационной сети «Интернет» и доступом в электронную информационно-образовательную среду образовательной организации	Браузер Google Chrome, Jupyter Notebook 6.3.0 и выше (язык Python с библиотеками Numpy, Pandas, gensim, NLTK, PyMorphy, фреймворком PyTorch)

Для самостоятельной работы обучающихся предусмотрены помещения, укомплектованные специализированной мебелью, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Наименование помещений для самостоятельной работы обучающихся	Оснащенность помещений для самостоятельной работы обучающихся	Перечень лицензионного программного обеспечения
Помещение для самостоятельной работы обучающихся (читальный зал Научной библиотеки)	Мебель: учебная мебель Комплект специализированной мебели: компьютерные столы Оборудование: компьютерная техника с подключением к информационно-коммуникационной сети «Интернет» и доступом в электронную информационно-образовательную среду образовательной организации, веб-камеры, коммуникационное оборудование,	MS Office Word 2016 и выше Ms Power Point 2016 и выше

	обеспечивающее доступ к сети интернет (проводное соединение и беспроводное соединение по технологии Wi-Fi)	
Помещение для самостоятельной работы обучающихся (Ауд. 101, 102, 103, 105/1, 106 и 106а)	Мебель: учебная мебель Комплект специализированной мебели: компьютерные столы Оборудование: компьютерная техника с подключением к информационно-коммуникационной сети «Интернет» и доступом в электронную информационно-образовательную среду образовательной организации, веб-камеры, коммуникационное оборудование, обеспечивающее доступ к сети интернет (проводное соединение и беспроводное соединение по технологии Wi-Fi)	Браузер Google Chrome, Matlab, Jupyter Notebook 6.3.0 и выше (язык Python с библиотеками Numpy, Pandas, gensim, NLTK, PyMorphy, фреймворком PyTorch)

№	Продукт	Параметры продукта	Кол-во	Кол-во конфигураций	Ед. изм.
1	Виртуальная машина	Виртуальная машина 10% vCPU 2 vCPU 4 RAM	1	60	Шт
		ОС Ubuntu 22.04	1		Шт
		Системный диск SSD	1		Шт
			10		Гб
		Аренда публичного IP	1		Шт
2	Виртуальная машина с GPU	Виртуальная машина с GPU NVIDIA® Tesla® V100 2 GPU 8 vCPU 128 Гб RAM	1	1	Шт
		ОС Ubuntu 24.04	1		Шт
		Системный диск SSD	1		Шт
			2000		Гб
		Диск SSD	1		Шт
			4096		Гб
		Диск SSD	1		Шт
			4096		Гб
		Аренда публичного IP	1		Шт
3	K8S	Master node 8 vCPU 16 RAM	1	1	Шт
		Worker node 10% доля 4 vCPU 32 RAM	5		Шт
		Worker node SSD-NVME	64		Гб
		Аренда публичного IP	1		Шт
4	ML Inference Instance Type GPU	Время работы в месяц	40	1	Ч
		Инстанс 8 x NVIDIA® H100 NVLink PCIe 160 vCPU 1520 GB RAM	1		Шт

		Количество запросов к ML-моделям	1		Млн. Шт
		Кэш ML-моделей	160		Гб
5	LLM	Токены GigaChat 2 Max	50		Млн. Шт
		Токены Embeddings	400		Млн. Шт