Автономная некоммерческая организация высшего образования
«Волгоградский институт бизнеса»

Рабочая программа учебной дисциплины

Проектирование систем с использованием технологий искусственного интеллекта

(Наименование дисциплины)

09.03.03 Прикладная информатика, направленность (профиль) «Прикладной искусственный интеллект»

(Направление подготовки / Профиль)

Бакалавр

(Квалификация)

Кафедра разработчик Экономики и управления

Год набора **2026**

Вид учебной деятельности	Трудоемкость (объем) дисциплины	
	Очная форма	Очно-заочная форма
	д	в
Зачетные единицы	**4**	**4**
Общее количество часов	144	144
Аудиторные часы контактной работы обучающегося с преподавателями:	36	16
– Лекционные (Л)	18	8
– Практические (ПЗ)	18	8
– Лабораторные (ЛЗ)		
– Семинарские (СЗ)		
Самостоятельная работа обучающихся (СРО)	108	128
К (Р-Г) Р (П) (+;-)		
Тестирование (+;-)		
ДКР (+;-)		
Зачет (+;-)	+	+
Зачет с оценкой (+;- (Кол-во часов))		
Экзамен (+;- (Кол-во часов))		

Волгоград 2026

Содержание

Раздел 1. Организационно-методический раздел

1.1. Цели освоения дисциплины

Дисциплина **«Проектирование систем с использованием технологий искусственного интеллекта»** входит **в перечень Обязательных дисциплин (модули) Б1.О.22** подготовки обучающихся по направлению **Прикладная информатика, направленность (профиль) «Прикладной искусственный интеллект».**

Целью дисциплины является формирование **компетенций** (*в соответствии с ФГОС ВО и требованиями к результатам освоения основной профессиональной образовательной программы высшего образования (ОПОП ВО)*):

ПК-3. Способен осуществлять проектирование компьютерного программного обеспечения;

Дескрипторы общепрофессиональной компетенции:

ПК-3.1. Способен проектировать архитектуру компьютерного программного обеспечения, включая интеллектуальные компоненты

ПК-3.2. Способен применять методы и средства проектирования программного обеспечения, включая проектирование интерфейсов и командную разработку

Перечисленные компетенции формируются в процессе достижения **индикаторов компетенций:**

Обобщенная трудовая функция/ трудовая функция	**Код и наименование дескриптора компетенций**	**Код и наименование индикатора достижения компетенций (из ПС)**
Тип задач проф. деятельности: ектный	ПК-3.1. Способен проектировать архитектуру компьютерного программного обеспечения, включая интеллектуальные компоненты ПК-3.2. Способен применять методы и средства проектирования программного обеспечения, включая проектирование интерфейсов и командную разработку	Знает: ИД-1 ПК 3.1. Принципы построения и виды архитектуры компьютерного программного обеспечения ИД-2 ПК 3.2. Методы и средства проектирования компьютерного программного обеспечения Умеет: ИД-3 ПК 3.1. Использовать существующие типовые решения и шаблоны проектирования компьютерного программного обеспечения D/03.6 ИД-4 ПК 3.2. Использовать командные средства разработки компьютерного программного обеспечения D/03.6 Имеет навыки и (или) опыт: ИД-5 ПК 3.1. Разработки, изменения архитектуры компьютерного программного обеспечения и ее согласования с системным аналитиком и архитектором программного обеспечения D/03.6 ИД-6 ПК 3.2. Проектирования программных интерфейсов D/03.6

1.2. Место дисциплины в структуре ОПОП ВО направления подготовки «09.03.03 Прикладная информатика», направленность (профиль) «Прикладной искусственный интеллект»

№	Предшествующие дисциплины (дисциплины, изучаемые параллельно)	Последующие дисциплины
1	*2*	*3*
1	Информационные технологии и искусственный интеллект	Выполнение и защита выпускной квалификационной работы
2	Компьютерная лингвистика	Производственная практика (Технологическая (проектно-технологическая) практика)
3	Базы данных	Производственная практика (Научно-исследовательская работа)
4	Гибридные системы поддержки принятия решений	Производственная практика (Преддипломная практика)
5	Компьютерное зрение	
6	Учебная практика (Технологическая (проектно-технологическая) практика)	
7	Учебная практика (Эксплуатационная практика)	

Последовательность формирования компетенций в указанных дисциплинах может быть изменена в зависимости от формы и срока обучения, а также преподавания с использованием дистанционных технологий обучения.

1.3. Нормативная документация

Рабочая программа учебной дисциплины составлена на основе:

- Федерального государственного образовательного стандарта высшего образования по направлению подготовки **09.03.03 Прикладная информатика**;
- Учебного плана направления подготовки **09.03.03 Прикладная информатика, направленность (профиль) «Прикладной искусственный интеллект»** 2026 года набора;
- Образца рабочей программы учебной дисциплины (приказ № 113-О от 01.09.2021 г.).

Раздел 2. Тематический план

Очная форма обучения (полный срок)

№	Тема дисциплины	Трудоемкость				
		Всего	Аудиторные занятия		СРО	Код индикатора и дескриптора достижения компетенций
			Л	ПЗ (ЛЗ, СЗ)		
1	*2*	*3*	*4*	*5*	*6*	*7*
1	Введение в проектирование ИИ-систем: жизненный цикл, отличие от традиционного ПО, роли в команде.	18	2	2	14	ИД-1 ПК- 3.1 ИД-2 ПК- 3.2
2	Сбор и подготовка данных: требования к качеству, разметка, маркировка, синтез данных.	18	2	2	14	ИД-2 ПК- 3.2 ИД-3 ПК- 3.1
3	Выбор архитектуры ИИ-компонентов: открытые модели, дообучение (fine-tuning), сжатие (дистилляция, квантизация).	18	2	2	14	ИД-3 ПК- 3.1 ИД-4 ПК- 3.2
4	Проектирование RAG-пайплайнов: векторизация, чанкинг, базы данных, ретриверы.	18	2	2	14	ИД-4 ПК- 3.2 ИД-5 ПК- 3.1
5	Интеграция ИИ в корпоративные системы: High-Level Architecture (монолит, микросервисы, serverless), создание API.	18	2	2	14	ИД-5 ПК- 3.1
6	Инженерные практики: CI/CD для ML, версионирование (DVC), контейнеризация (Docker), мониторинг.	18	2	2	14	ИД-5 ПК- 3.1
7	Обеспечение безопасности и надежности: промпт-инженерия, защита от инъекций, фильтрация, асессменты.	16	2	2	12	ИД-5 ПК- 3.1 ИД-6 ПК- 3.2
8	Проектирование взаимодействия: Human-in-the-Loop (HITL), объяснимость (XAI), UX для генеративных систем.	20	4	4	12	ИД-5 ПК- 3.1 ИД-6 ПК- 3.2
Вид промежуточной аттестации (Зачет)		+				
Итого		**144**	**18**	**18**	**108**	

Очно-заочная форма обучения (полный срок)

№	Тема дисциплины	Трудоемкость				
		Всего	Аудиторные занятия		СРО	Код индикатора и дескриптора достижения компетенций
			Л	ПЗ (ЛЗ, СЗ)		
1	*2*	*3*	*4*	*5*	*6*	*7*
1	Введение в проектирование ИИ-систем: жизненный цикл, отличие от традиционного ПО, роли в команде.	18	2		16	ИД-1 ПК- 3.1 ИД-2 ПК- 3.2
2	Сбор и подготовка данных: требования к качеству, разметка, маркировка, синтез данных.	18		2	16	ИД-2 ПК- 3.2 ИД-3 ПК- 3.1

3	Выбор архитектуры ИИ-компонентов: открытые модели, до-обучение (fine-tuning), сжатие (ди-стилляция, квантизация).	18		2	16	ИД-3 ПК- 3.1 ИД-4 ПК- 3.2
4	Проектирование RAG-пайплайнов: векторизация, чанкинг, базы данных, ретриверы.	18	2		16	ИД-4 ПК- 3.2 ИД-5 ПК- 3.1
5	Интеграция ИИ в корпоративные системы: High-Level Architecture (моно-лит, микросервисы, serverless), созда-ние API.	18	2		16	ИД-5 ПК- 3.1
6	Инженерные практики: CI/CD для ML, версионирование (DVC), контей-неризация (Docker), мониторинг.	18		2	16	ИД-5 ПК- 3.1
7	Обеспечение безопасности и надеж-ности: промпт-инженерия, защита от инъекций, фильтрация, асессменты.	18		2	16	ИД-5 ПК- 3.1 ИД-6 ПК- 3.2
8	Проектирование взаимодействия: Human-in-the-Loop (HITL), объясни-мость (XAI), UX для генеративных систем.	18	2		16	ИД-5 ПК- 3.1 ИД-6 ПК- 3.2
Вид промежуточной аттестации (Зачет)		+				
Итого		**144**	**8**	**8**	**128**	

Раздел 3. Содержание дисциплины

3.1. Содержание дисциплины

Тема 1. Введение в проектирование ИИ-систем: жизненный цикл, отличие от традиционного ПО, роли в команде

Лекция: Определение систем с использованием искусственного интеллекта (ИИ-систем) как программных продуктов, включающих компоненты машинного обучения, обработки естественного языка или генеративных моделей. Ключевое отличие ИИ-систем от традиционных: наличие недетерминированного поведения (вероятностный вывод, изменчивость качества), зависимость от обучающих данных, необходимость непрерывного мониторинга в продуктивной среде. Жизненный цикл ИИ-систем: бизнес-анализ и метрики успеха, сбор и разметка данных, выбор архитектуры, обучение и валидация, развёртывание, мониторинг и обновление. Сравнение с классическим Waterfall и Agile. Роли в команде: Product Owner (бизнес-цели), Data Scientist (модели), ML Engineer (развёртывание и пайплайны), Data Engineer (данные), AI Architect (интеграция). Понятие технического долга в ML (ML Technical Debt). Примеры: внедрение чат-бота поддержки, системы рекомендаций, голосового ассистента.

Тема 2. Сбор и подготовка данных: требования к качеству, разметка, маркировка, синтез данных

Лекция: Данные как ключевой компонент ИИ-системы. Принцип GIGO (Garbage In, Garbage Out) и его влияние на проектирование. Требования к качеству данных: полнота, точность, репрезентативность, актуальность, непротиворечивость. Разметка данных (аннотирование): ручная, краудсорсинг, автоматическая (разметка с помощью слабых моделей, правил). Инструменты разметки: LabelStudio, CVAT, SuperAnnotate. Проблемы разметки: субъективность, ошибки экспертов, стоимость. Маркировка конфиденциальных данных и требования GDPR, ФЗ-152. Синтез данных: аугментация (повороты, шумы, цветовые трансформации), генерация синтетических данных с помощью GAN или LLM. Применение синтетических данных для редких классов или при нехватке реальных данных. Документирование датасетов: паспорт данных, происхождение (data

lineage), контроль версий (DVC, Hugging Face Datasets). Проектирование Data Lake и Feature Store.

Тема 3. Выбор архитектуры ИИ-компонентов: открытые модели, дообучение, сжатие

Лекция: Стратегии использования моделей: Zero-shot/Inference-only (использование готовых API), Few-shot промптинг, Fine-tuning (дообучение под задачу), Full-training (обучение с нуля). Открытые модели (Open Source): требования к лицензиям, сообщество, ограничения. Платформы моделей: Hugging Face, Ollama, TensorFlow Hub, PyTorch Hub. Выбор размера модели: trade-off между качеством и латентностью/стоимостью. Дообучение (fine-tuning) как основной метод адаптации: полное дообучение, LoRA (Low-Rank Adaptation), QLoRA (Quantized LoRA). Параметроэффективные методы (PEFT). Сжатие моделей: дистилляция знаний (обучение малой модели имитировать большую), квантизация (снижение точности весов INT8/INT4), прунинг (удаление незначимых весов). Выбор архитектуры: изолированный сервис моделей vs встраивание в монолит. Проектирование Model Registry и версионирование (MLflow, Weights & Biases).

Тема 4. Проектирование RAG-пайплайнов: векторизация, чанкинг, базы данных, ретриверы

Лекция: RAG (Retrieval-Augmented Generation) как архитектурный паттерн для генеративных ИИ-систем. Компоненты пайплайна: эмбеддинг-модель (получение векторов), векторная база данных (поиск), ретривер (ранжирование), LLM (генерация ответа с контекстом). Этапы проектирования: индексация (подготовка знаний), поиск (ретривер) и генерация. Чанкинг (разбиение документов): стратегии (semantic chunking, fixed-size, recursive), выбор размера и перекрытия чанков. Векторизация: эмбеддинги (OpenAI, Cohere, BERT, LaBSE), мультимодальные эмбеддинги (CLIP). Векторные базы данных (Vector DB): FAISS (библиотека), Chroma (легковесная), Qdrant, Pinecone, Weaviate. Индексация: HNSW (иерархический граф для ANN), Flat (точный поиск). Ретриверы: dense passage retrieval (DPR), hybrid search (векторный+ключевой), переранжирование (cross-encoder). Оценка RAG: метрики релевантности (Hit Rate, MRR, NDCG), точность ответов (наличие галлюцинаций). Продвинутые техники: self-querying, RAPTOR, графовые RAG.

Тема 5. Интеграция ИИ в корпоративные системы: High-Level Architecture, создание API

Лекция: Место ИИ-компонентов в общей архитектуре предприятия. High-Level Architecture (HLA): диаграмма уровней (пользовательский интерфейс, сервисы, модели, данные). Сценарии интеграции: синхронный (REST/gRPC) для критичного ко времени ответа, асинхронный (очереди сообщений: RabbitMQ, Kafka) для тяжелых/долгих задач, батч-инференс для аналитики. Проектирование Model Serving: in-process (библиотека внутри приложения), микросервис (FastAPI, Django), serverless (AWS Lambda, Cloud Functions), dedicated serving (TensorFlow Serving, TorchServe, Triton). Создание API для ИИ-моделей: спецификация (OpenAPI), версионирование эндпоинтов (v1, v2), валидация входных данных (Pydantic, Cerberus), rate limiting, аутентификация (API keys, OAuth). Проектирование Gateway для ИИ: маршрутизация запросов к разным провайдерам, кэширование, fallback-стратегии при ошибках модели. Выбор протоколов: REST (JSON), gRPC (protobuf, высокая производительность), WebSocket (потоковая передача/чат). Интеграция с брокерами данных (Kafka, AWS Kinesis) для событийного взаимодействия.

Тема 6. Инженерные практики: CI/CD для ML, версионирование, контейнеризация, мониторинг

Лекция: Адаптация DevOps практик для ML: MLOps. CI/CD для ML: непрерывная интеграция кода и данных, непрерывная доставка моделей. Компоненты CI/CD пайплайна: сборка кода (линтеры, тесты), проверка данных (валидация схемы, обнаружение выбросов), обучение и оценка модели (автоматический прогон экспериментов), регистрация модели, деплой на staging/production. Версионирование: Git (код), DVC/Data Version Control (данные, артефакты), MLflow (модели и эксперименты). Контейнеризация: Dockerfile для моделей, оптимизация слоев, минимизация образа. Оркестрация: Kubernetes, Docker Compose; управление GPU в K8S. Мониторинг (Observability): метрики качества модели (дрейф данных и концепта — PSI), системные метрики (latency, throughput, ошибки), алертинг и дашборды (Prometheus, Grafana). Логирование пред-

сказаний. Тестирование в ML: юнит-тесты (препроцессинг, метрики), интеграционное тестирование (API), A/B тестирование, канареечный деплой.

Тема 7. Обеспечение безопасности и надежности: промпт-инженерия, защита от инъекций, фильтрация, асессменты

Лекция: Специфика безопасности ИИ-систем: уязвимости LLM и ML моделей (OWASP Top 10 for LLM). Основные угрозы: Prompt Injection (прямая/косвенная) — игнорирование системного промпта, Data Poisoning (отравление данных), Model Inversion (восстановление данных из модели), Membership Inference (определение наличия данных в обучении). Стратегии защиты: валидация и санитизация пользовательского ввода, изоляция LLM, разделение системного и пользовательского контента (XML теги, специфичные разделители). Фильтрация ввода/вывода: регулярные выражения, персональные данные (PII), нецензурная лексика, запрещенные темы. Промпт-инженерия для безопасности: явные инструкции по отклонению вредоносных запросов (defensive prompts). Роли и системы: использование модерационных моделей (фильтрация OpenAI Moderation API), паттерны на основе правил (Guardrails). Асессменты: красные команды (red teaming), соревнования по взлому промптов, пентест ИИ-систем. Обеспечение надежности: контрактное тестирование (expectations), graceful fallback при отказе модели, рейтинг уверенности (confidence score), человеческий контроль (HITL) для критичных действий.

Тема 8. Проектирование взаимодействия: Human-in-the-Loop (HITL), объяснимость (XAI), UX для генеративных систем

Лекция: Принципы Human-in-the-Loop (HITL): человек как контролёр качества, поставщик обратной связи, источник данных для дообучения. Степени вовлечения: минимальная (автоматический режим с аудитом), средняя (запрос подтверждения перед действиями), полная (человек принимает решение, ИИ — ассистент). Проектирование интерфейсов для подтверждения: кнопки «Принять/Отклонить», возможность редактирования сгенерированных ответов. Объяснимость (XAI): тенденция к интерпретируемости при принятии решений. Применение методов: SHAP, LIME, важность признаков. Объяснение работы RAG: отображение источников. UX для генеративных систем: Управление ожиданиями времени ответа (индикаторы загрузки, стриминг). Демонстрация уверенности (калибровка) и сомнений модели. Поддержка цикла обратной связи (лайки, комментарии, исправления). Этика и конфиденциальность: прозрачное уведомление пользователя о том, что он общается с ИИ, протоколы получения согласия на сбор данных для обучения.

3.2. Содержание практического блока дисциплины

Очная форма обучения (полный срок)

№	Тема практического (семинарского, лабораторного) занятия
1	***2***
ПЗ 1	Практическое занятие: Анализ кейса реальной ИИ-системы (например, чат-поддержки или системы предиктивной аналитики). Определение этапов жизненного цикла для кейса. Составление карты стейкхолдеров и матрицы RACI (роли: Data Scientist, ML Engineer, Data Engineer, DevOps, бизнес-заказчик). Разработка чек-листа действий на фазе бизнес-анализа. Вычисление метрик успеха проекта (KPI) и их перевод в технические требования к модели (confusion matrix cost matrix). Проектирование диаграммы потока данных и управления моделями (блок-схема) с использованием блоков (сбор данных — обучение — деплой — мониторинг). Моделирование ситуации «падения качества в проде»: brainstorm возможных причин и составление плана действий.
ПЗ 2	Практическое занятие: Работа с реальным «грязным» датасетом (например, отзывы клиентов). Проведение профилирования данных: расчёт полноты, уникальности, дубликатов, статистик для обнаружения выбросов. Создание паспорта данных (data catalog) — фиксация источников, типов, распределений. Проектирование схемы разметки: создание инструкции для аннотатора (labeling guidelines) для конкретной за-

	дачи (NER, классификация). Реализация простейшего пайплайна аугментации на Python (например, back-translation, замена синонимов, поворот изображений) и оценка влияния аугментации на небольшой модели. Вычисление стоимости синтеза данных при заказе у сервиса разметки и оценка бюджета проекта.
ПЗ 3	Практическое занятие: Поиск и сравнение открытых моделей на Hugging Face для заданной задачи (например, суммаризация текста). Оценка моделей по критериям: размер (параметры), лицензия (Apache 2.0, MIT), частота загрузок (downloads), качество на бенчмарках (ROUGE, BLEU). Разработка архитектурного эскиза (компонент моделирования) для системы. Дообучение (fine-tuning) небольшой модели BERT (DistilBERT) в Google Colab с использованием трансформеров. Эксперимент: сравнение времени инференса и потребления памяти для базовой модели и её квантизированной версии (INT8) с использованием библиотеки Optimum или bitsandbytes. Расчет вычислительных затратов на дообучение и стоимости железа/облака.
ПЗ 4	Практическое занятие: Реализация прототипа RAG с помощью LangChain или LlamaIndex. Загрузка набора документов (корпоративный wiki или статьи). Эксперименты со стратегиями чанкинга (фиксированный размер, семантическое разбиение, перекрытие). Генерация эмбеддингов с помощью модели sentence-transformers. Создание векторной базы данных (Chroma или FAISS). Реализация поиска с использованием ретривера (например, parent-document retriever) и переранжировщика (cross-encoder). Оценка качества поиска (retrieval) и полноты ответа. Сравнение «наивного» RAG с улучшенным Self-querying RAG. Визуализация найденных чанков и источников ответа.
ПЗ 5	Практическое занятие: Проектирование High-Level Architecture (HLA) в Draw.io или аналогичном инструменте. Разработка двух вариантов интеграции: синхронный REST-сервер на FastAPI (endpoint /chat) и асинхронный обработчик через очереди (Celery + Redis). Определение контракта API (OpenAPI/Swagger) для синхронного взаимодействия: схемы запроса и ответа. Реализация (кодирование) простого API-сервиса, обертывающего модель машинного обучения. Реализация клиента (CLI скрипт или Postman) для тестирования. Проведение нагрузочного тестирования (locust) для оценки пропускной способности (RPS) полученного сервиса. Расчет задержек (latency) при разных типах инференса.
ПЗ 6	Практическое занятие: Создание репозитория Git со структурой ML-проекта. Инициализация DVC и подключение удаленного хранилища (S3, GDrive). Настройка GitHub Actions CI/CD пайплайна. Написание Dockerfile для сервиса инференса. Использование MLflow для логирования экспериментов (loss, accuracy) и регистрации моделей. Развертывание модели (деплой) в Docker контейнере на локальной машине. Настройка мониторинга (Prometheus + Grafana) для отслеживания количества запросов и времени ответа. Расчет PSI (Population Stability Index) для детекции дрейфа на новых данных.
ПЗ 7	Практическое занятие: Эксперименты с промпт инъекциями: составление запросов, пытающихся обойти системные инструкции (игнорирование правил, раскрытие промпта). Разработка защитных мер (defensive prompts) и оценка их эффективности. Реализация простого фильтра на проверку персональных данных (PII) для исходящего текста. Создание системы guardrails (оценка и модерация ответов). Сравнение разных подходов к оценке токсичности (toxic-bert, OpenAI moderation). Тестирование устойчивости модели к adversarial атакам (для текстовых моделей). Проведение ролевой игры «Red Team vs Blue Team»: одна группа пытается взломать промпт, другая — защитить.
ПЗ 8	Практическое занятие: Проектирование интерфейса для модерации или подтверждения действия (HITL). Разработка прототипа интерфейса в Figma или на Streamlit: показ фактов (источников) RAG, оценка (лайк/дизлайк), возможность редактирования ответа. Применение библиотеки Captum или SHAP для объяснения предсказаний модели на конкретном примере (создание force plot). Анализ важности признаков и составление отчета об объяснимости. Опрос пользователей (симуляция) для сбора об-

	ратной связи по качеству интерфейса и контента. Проектирование цикла обратной связи: как лайки и исправления пользователей превращаются в обучающие данные для дообучения.

Очно-заочная форма обучения (полный срок)

№	Тема практического (семинарского, лабораторного) занятия
1	***2***
ПЗ 2	Практическое занятие: Работа с реальным «грязным» датасетом (например, отзывы клиентов). Проведение профилирования данных: расчёт полноты, уникальности, дубликатов, статистик для обнаружения выбросов. Создание паспорта данных (data catalog) — фиксация источников, типов, распределений. Проектирование схемы разметки: создание инструкции для аннотатора (labeling guidelines) для конкретной задачи (NER, классификация). Реализация простейшего пайплайна аугментации на Python (например, back-translation, замена синонимов, поворот изображений) и оценка влияния аугментации на небольшой модели. Вычисление стоимости синтеза данных при заказе у сервиса разметки и оценка бюджета проекта.
ПЗ 3	Практическое занятие: Поиск и сравнение открытых моделей на Hugging Face для заданной задачи (например, суммаризация текста). Оценка моделей по критериям: размер (параметры), лицензия (Apache 2.0, MIT), частота загрузок (downloads), качество на бенчмарках (ROUGE, BLEU). Разработка архитектурного эскиза (компонент моделирования) для системы. Дообучение (fine-tuning) небольшой модели BERT (DistilBERT) в Google Colab с использованием трансформеров. Эксперимент: сравнение времени инференса и потребления памяти для базовой модели и её квантизированной версии (INT8) с использованием библиотеки Optimum или bitsandbytes. Расчет вычислительных затратов на дообучение и стоимости железа/облака.
ПЗ 6	Практическое занятие: Создание репозитория Git со структурой ML-проекта. Инициализация DVC и подключение удаленного хранилища (S3, GDrive). Настройка GitHub Actions CI/CD пайплайна. Написание Dockerfile для сервиса инференса. Использование MLflow для логирования экспериментов (loss, accuracy) и регистрации моделей. Развертывание модели (деплой) в Docker контейнере на локальной машине. Настройка мониторинга (Prometheus + Grafana) для отслеживания количества запросов и времени ответа. Расчет PSI (Population Stability Index) для детекции дрейфа на новых данных.
ПЗ 7	Практическое занятие: Эксперименты с промпт инъекциями: составление запросов, пытающихся обойти системные инструкции (игнорирование правил, раскрытие промпта). Разработка защитных мер (defensive prompts) и оценка их эффективности. Реализация простого фильтра на проверку персональных данных (PII) для исходящего текста. Создание системы guardrails (оценка и модерация ответов). Сравнение разных подходов к оценке токсичности (toxic-bert, OpenAI moderation). Тестирование устойчивости модели к adversarial атакам (для текстовых моделей). Проведение ролевой игры «Red Team vs Blue Team»: одна группа пытается взломать промпт, другая — защитить.

3.3. Образовательные технологии

Очная форма обучения (полный срок)

№	Тема занятия	Вид учебного занятия	Форма / Методы интерактивного обучения	% учебного времени
1	*2*	*3*	*4*	*5*
1	Введение в проектирование ИИ-систем: жизненный цикл, отличие от традиционного ПО, роли в команде.	ПЗ	Проектно-ориентированное обучение, Кейс-стади (Case Study), Дискуссионные технологии	25
2	Сбор и подготовка данных: требования к качеству, разметка, маркировка, синтез данных.	ПЗ	Кейс-стади (Case Study) и анализ реальных ситуаций, Инструментальные технологии	25
3	Выбор архитектуры ИИ-компонентов: открытые модели, дообучение (fine-tuning), сжатие (дистилляция, квантизация).	ПЗ	Инструментальные технологии, Проектно-ориентированное обучение	25
4	Проектирование RAG-пайплайнов: векторизация, чанкинг, базы данных, ретриверы.	ПЗ	Инструментальные технологии, Проектно-ориентированное обучение	25
5	Интеграция ИИ в корпоративные системы: High-Level Architecture (монолит, микросервисы, serverless), создание API.	ПЗ	Инструментальные технологии, Проектно-ориентированное обучение, Технология контекстного обучения	25
6	Инженерные практики: CI/CD для ML, версионирование (DVC), контейнеризация (Docker), мониторинг.	ПЗ	Инструментальные технологии, Проектно-ориентированное обучение, Технология контекстного обучения	25
7	Обеспечение безопасности и надежности: промпт-инженерия, защита от инъекций, фильтрация, асессменты.	ПЗ	Инструментальные технологии, Проектно-ориентированное обучение, Технология контекстного обучения	50
8	Проектирование взаимодействия: Human-in-the-Loop (HITL), объяснимость (XAI), UX для генеративных систем.	ПЗ	Инструментальные технологии, Проектно-ориентированное обучение, Технология контекстного обучения	25

Очно-заочная форма обучения (полный срок)

№	Тема занятия	Вид учебного занятия	Форма / Методы интерактивного обучения	% учебного времени
1	*2*	*3*	*4*	*5*
2	Сбор и подготовка данных: требования к качеству, разметка, маркировка, синтез данных.	ПЗ	Кейс-стади (Case Study) и анализ реальных ситуаций, Инструментальные технологии	25

3	Выбор архитектуры ИИ-компонентов: открытые модели, до-обучение (fine-tuning), сжатие (ди-стилляция, квантизация).	ПЗ	Инструментальные технологии, Проектно-ориентированное обу-чение	25
6	Инженерные практики: CI/CD для ML, версионирование (DVC), кон-тейнеризация (Docker), мониторинг.	ПЗ	Инструментальные технологии, Проектно-ориентированное обу-чение, Технология кон-текстного обучения	25
7	Обеспечение безопасности и надеж-ности: промпт-инженерия, защита от инъекций, фильтрация, асессменты.	ПЗ	Инструментальные технологии, Проектно-ориентированное обу-чение, Технология кон-текстного обучения	50

Раздел 4. Организация самостоятельной работы обучающихся

4.1. Организация самостоятельной работы обучающихся

№	Тема дисциплины	№ вопросов	№ рекомендуемой литературы
1	*2*	*3*	*4*
1	Введение в проектирование ИИ-систем: жизненный цикл, отличие от традиционного ПО, роли в команде.	1-5	2, 3, 4, 10, 15
2	Сбор и подготовка данных: требования к качеству, разметка, маркировка, синтез данных.	6-10	7, 8, 9, 13
3	Выбор архитектуры ИИ-компонентов: открытые модели, дообучение (fine-tuning), сжатие (дистилляция, квантизация).	11-15	5, 7, 12, 13
4	Проектирование RAG-пайплайнов: векторизация, чанкинг, базы данных, ретриверы.	16-20	1, 2, 6, 11, 14, 15
5	Интеграция ИИ в корпоративные системы: High-Level Architecture (монолит, микросервисы, serverless), создание API.	21-25	1, 2, 6, 11, 14, 15
6	Инженерные практики: CI/CD для ML, версионирование (DVC), контейнеризация (Docker), мониторинг.	26-30	7
7	Обеспечение безопасности и надежности: промпт-инженерия, защита от инъекций, фильтрация, асессменты.	31-35	5, 6
8	Проектирование взаимодействия: Human-in-the-Loop (HITL), объяснимость (XAI), UX для генеративных систем.	36-40	2, 4, 10, 12, 14, 15

Перечень вопросов, выносимых на самостоятельную работу обучающихся

1. Перечислите ключевые отличия жизненного цикла ИИ-системы от традиционного ПО.
2. Какие роли входят в команду разработки ИИ-системы и за что отвечает каждая из них?
3. Как перевести бизнес-метрики успеха в технические требования к модели машинного обучения?
4. Что такое технический долг (Technical Debt) в контексте машинного обучения и какие его виды существуют?
5. Почему качество данных критически важно для успеха ИИ-проекта? Объясните принцип GIGO.
6. Что входит в процесс профилирования данных? Какие метрики качества данных проверяются?
7. Какие существуют стратегии разметки данных? Сравните ручную, автоматическую и краудсорсинговую разметку.
8. Что такое синтез данных (data synthesis) и в каких случаях он применяется?
9. Опишите разницу между стратегиями использования ИИ-моделей: Zero-shot, Few-shot, Fine-tuning и Full-training.
10. Назовите критерии выбора между открытой и проприетарной моделью при проектировании системы.
11. Что такое параметро-эффективное дообучение (PEFT)? Опишите метод LoRA.
12. Какие методы сжатия моделей вы знаете? Опишите дистилляцию и квантизацию.
13. Перечислите основные компоненты RAG-пайплайна.
14. Какие существуют стратегии чанкинга (разбиения) документов в RAG? Влияние размера чанка на качество поиска.
15. Для чего используются векторные базы данных? Приведите примеры популярных вектор-

ных БД и их особенности.
16. Чем отличается dense retriever от sparse retriever в контексте поиска информации?
17. Какие бывают сценарии интеграции ИИ-моделей в корпоративные системы?
18. Опишите разницу между синхронным и асинхронным инференсом. Когда какой подход предпочтительнее?
19. Что такое Model Serving и какие существуют способы развертывания моделей?
20. Какие требования предъявляются к API (REST/gRPC) для ИИ-моделей?
21. В чем заключается суть MLOps? Какие задачи решает CI/CD пайплайн для ML?
22. Какие инструменты используются для версионирования данных и моделей?
23. Зачем нужна контейнеризация (Docker) в ML-проектах и почему это важно для воспроизводимости?
24. Какие метрики необходимо отслеживать при мониторинге ИИ-системы в продуктивной среде?
25. Что такое дрейф данных (Data Drift) и дрейф концепта (Concept Drift)? Как их обнаружить?
26. Какие уязвимости входят в OWASP Top 10 для LLM? Перечислите основные.
27. Что такое промпт-инъекция (Prompt Injection) и как защититься от этой атаки?
28. Какие методы фильтрации ввода и вывода используются для обеспечения безопасности ИИ-систем?
29. Что такое Red Teaming в контексте ИИ-безопасности?
30. Раскройте концепцию Human-in-the-Loop (HITL). Приведите примеры степеней вовлечения человека.
31. Чем объяснимость модели (XAI) отличается от интерпретируемости?
32. Какие существуют методы визуализации важности признаков для объяснения решений моделей?
33. Каковы основные принципы проектирования пользовательского опыта (UX) для генеративных ИИ-систем?
34. Как в RAG-системе можно визуализировать источники ответа для повышения доверия пользователя?
35. Как организовать цикл сбора обратной связи от пользователя для дообучения модели?
36. Какие требования предъявляются к ИИ-системам согласно законодательству о персональных данных?
37. Как оценить эффективность RAG-пайплайна? Какие метрики для этого используются?
38. В чем разница между A/B тестированием и канареечным деплоем (canary deployment) для ИИ-моделей?
39. Какие подходы к тестированию ИИ-систем существуют? Чем модульное тестирование отличается от контрактного?
40. Спроектируйте (опишите словами или схемой) архитектуру простой RAG-системы.

4.2. Перечень учебно-методического обеспечения самостоятельной работы обучающихся

Самостоятельная работа обучающихся обеспечивается следующими учебно-методическими материалами:

1. Указаниями в рабочей программе по дисциплине (п.4.1.)
2. Лекционные материалы в составе учебно-методического комплекса по дисциплине
3. Заданиями и методическими рекомендациями по организации самостоятельной работы обучающихся в составе учебно-методического комплекса по дисциплине.
4. Глоссарием по дисциплине в составе учебно-методического комплекса по дисциплине.

Раздел 5. Фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся

Фонд оценочных средств по дисциплине представляет собой совокупность контролирующих материалов, предназначенных для измерения уровня достижения обучающимися установленных результатов образовательной программы. ФОС по дисциплине используется при проведении оперативного контроля и промежуточной аттестации обучающихся. Требования к структуре и содержанию ФОС дисциплины регламентируются Положением о фонде оценочных материалов по программам высшего образования – программам бакалавриата, магистратуры.

5.1. Паспорт фонда оценочных средств

Очная форма обучения (полный срок)

№	Контролируемые разделы (темы) дисциплины	Оценочные средства			
		Л	ПЗ (ЛЗ, СЗ)	СРО	Код индикатора и дескриптора достижения компетенций
1	*2*	*3*	*4*	*5*	*6*
1	Введение в проектирование ИИ-систем: жизненный цикл, отличие от традиционного ПО, роли в команде.	УО	ЗЗ, МШ	ПРВ	ИД-1 ПК- 3.1 ИД-2 ПК- 3.2
2	Сбор и подготовка данных: требования к качеству, разметка, маркировка, синтез данных.	УО	ЗЗ, Д	ПРВ	ИД-2 ПК- 3.2 ИД-3 ПК- 3.1
3	Выбор архитектуры ИИ-компонентов: открытые модели, дообучение (fine-tuning), сжатие (дистилляция, квантизация).	УО	ЗЗ, Д, МШ	ПРВ	ИД-3 ПК- 3.1 ИД-4 ПК- 3.2
4	Проектирование RAG-пайплайнов: векторизация, чанкинг, базы данных, ретриверы.	УО	ЗЗ, Д, МП	ПРВ	ИД-4 ПК- 3.2 ИД-5 ПК- 3.1
5	Интеграция ИИ в корпоративные системы: High-Level Architecture (монолит, микросервисы, serverless), создание API.	УО	ЗЗ, МШ	ПРВ	ИД-5 ПК- 3.1
6	Инженерные практики: CI/CD для ML, версионирование (DVC), контейнеризация (Docker), мониторинг.	УО	ЗЗ, Д	ПРВ	ИД-5 ПК- 3.1
7	Обеспечение безопасности и надежности: промпт-инженерия, защита от инъекций, фильтрация, асессменты.	УО	ЗЗ, Д, МШ	ПРВ	ИД-5 ПК- 3.1 ИД-6 ПК- 3.2
8	Проектирование взаимодействия: Human-in-the-Loop (HITL), объяснимость (XAI), UX для генеративных систем.	УО	ЗЗ, Д, МП	ПРВ	ИД-5 ПК- 3.1 ИД-6 ПК- 3.2

Очно-заочная форма обучения (полный срок)

№	Контролируемые разделы (темы) дисциплины	Оценочные средства			
		Л	ПЗ (ЛЗ, СЗ)	СРО	Код индикатора и дескриптора достижения компетенций
1	*2*	*3*	*4*	*5*	*6*

1	Введение в проектирование ИИ-систем: жизненный цикл, отличие от традиционного ПО, роли в команде.	УО	ЗЗ, МШ	ПРВ	ИД-1 ПК- 3.1 ИД-2 ПК- 3.2
2	Сбор и подготовка данных: требования к качеству, разметка, маркировка, синтез данных.	УО	ЗЗ, Д	ПРВ	ИД-2 ПК- 3.2 ИД-3 ПК- 3.1
3	Выбор архитектуры ИИ-компонентов: открытые модели, дообучение (fine-tuning), сжатие (дистилляция, квантизация).	УО	ЗЗ, Д, МШ	ПРВ	ИД-3 ПК- 3.1 ИД-4 ПК- 3.2
4	Проектирование RAG-пайплайнов: векторизация, чанкинг, базы данных, ретриверы.	УО	ЗЗ, Д, МП	ПРВ	ИД-4 ПК- 3.2 ИД-5 ПК- 3.1
5	Интеграция ИИ в корпоративные системы: High-Level Architecture (монолит, микросервисы, serverless), создание API.	УО	ЗЗ, МШ	ПРВ	ИД-5 ПК- 3.1
6	Инженерные практики: CI/CD для ML, версионирование (DVC), контейнеризация (Docker), мониторинг.	УО	ЗЗ, Д	ПРВ	ИД-5 ПК- 3.1
7	Обеспечение безопасности и надежности: промпт-инженерия, защита от инъекций, фильтрация, асессменты.	УО	ЗЗ, Д, МШ	ПРВ	ИД-5 ПК- 3.1 ИД-6 ПК- 3.2
8	Проектирование взаимодействия: Human-in-the-Loop (HITL), объяснимость (XAI), UX для генеративных систем.	УО	ЗЗ, Д, МП	ПРВ	ИД-5 ПК- 3.1 ИД-6 ПК- 3.2

Условные обозначения оценочных средств (Столбцы 3, 4, 5):

ЗЗ – защита выполненных заданий (творческих, расчетных и т.д.), представление презентаций;

ПРВ – проверка рефератов, отчетов, рецензий, аннотаций, конспектов, графического материала, эссе, переводов, решений заданий, выполненных заданий в электронном виде и т.д.;

МШ – Метод мозгового штурма;

Д – Дискуссия, полемика, диспут, дебаты;

МП – Метод проектов.

5.2. Тематика письменных работ обучающихся

1. Анализ жизненного цикла ИИ-продукта: от бизнес-идеи до мониторинга в продакшене (на примере конкретной системы).
2. Сравнительный анализ методологий управления проектами (Waterfall, Agile, CRISP-DM) при разработке ИИ-систем.
3. Проектирование процесса сбора и разметки данных для задачи компьютерного зрения (на примере датасета).
4. Применение методов аугментации и синтеза данных для балансировки выборки в задачах классификации.
5. Выбор архитектуры ИИ-компонента: сравнительный анализ открытых моделей для задачи обработки естественного языка.
6. Исследование эффективности методов параметро-эффективного дообучения (LoRA, QLoRA) для адаптации LLM под предметную область.
7. Оценка влияния методов сжатия моделей (квантизация, дистилляция, прунинг) на латентность и точность инференса.
8. Проектирование RAG-пайплайна: сравнение стратегий чанкинга и выбора эмбеддинг-модели для корпоративной базы знаний.
9. Разработка прототипа вопросно-ответной системы на основе RAG с использованием

LangChain и Chroma.
10. Проектирование High-Level Architecture системы с ИИ-компонентами на основе микросервисной архитектуры.
11. Разработка и нагрузочное тестирование REST API для сервиса генеративных моделей (FastAPI, Docker).
12. Внедрение практик MLOps: построение CI/CD пайплайна для автоматического обучения и деплоя модели.
13. Контейнеризация и оркестрация ML-сервисов: разработка Dockerfile и манифестов Kubernetes для приложения.
14. Проектирование системы мониторинга для детекции дрейфа данных и концепта в продуктивной среде.
15. Анализ уязвимостей LLM (OWASP Top 10) и разработка защитных механизмов против промпт-инъекций.
16. Разработка системы guardrails для фильтрации конфиденциальных данных (PII) в генеративных системах.
17. Проектирование пользовательского интерфейса с элементами Human-in-the-Loop для системы модерации контента.
18. Применение методов объяснимого ИИ (SHAP, LIME) для интерпретации решений модели в кредитном скоринге.
19. Проектирование цикла обратной связи для дообучения модели на основе пользовательских исправлений.
20. Сравнительный анализ векторных баз данных (FAISS, Chroma, Qdrant) для задач поиска в RAG-системах.
21. Организация версионирования данных и моделей с использованием DVC и MLflow в проекте машинного обучения.
22. Оценка экономической эффективности внедрения ИИ-системы: расчёт ROI, TCO и анализ затрат на инфраструктуру.
23. Разработка стратегии A/B тестирования для сравнения двух версий рекомендательной модели в продакшене.
24. Проектирование аварийно-устойчивой архитектуры ИИ-сервиса: fallback-стратегии igraceful degradation.
25. Этические аспекты проектирования ИИ-систем: предотвращение смещений (bias), обеспечение прозрачности и справедливости решений.

5.3. Перечень вопросов промежуточной аттестации по дисциплине

Вопросы к зачету:

1. Перечислите ключевые отличия жизненного цикла ИИ-системы от традиционного ПО. Какие этапы добавляются?
2. Какие роли входят в команду разработки ИИ-системы? Опишите зоны ответственности Data Scientist, ML Engineer и Data Engineer.
3. Что такое технический долг (Technical Debt) в контексте машинного обучения? Приведите примеры.
4. Как перевести бизнес-метрики успеха в технические требования к модели? Приведите пример.
5. Какие требования предъявляются к качеству данных в ИИ-проектах? Что такое принцип GIGO?
6. Опишите основные стратегии разметки данных (ручная, автоматическая, краудсорсинг). Каковы их преимущества и недостатки?
7. Что такое синтез данных (data synthesis)? В каких случаях он применяется и какие методы используются?
8. Сравните стратегии использования ИИ-моделей: Zero-shot, Few-shot, Fine-tuning и Full-training. Когда какая стратегия предпочтительнее?

9. Что такое параметро-эффективное дообучение (PEFT)? Опишите метод LoRA и его преимущества.
10. Какие методы сжатия моделей вы знаете? Опишите дистилляцию знаний и квантизацию.
11. Перечислите основные компоненты RAG-пайплайна и объясните их назначение.
12. Какие существуют стратегии чанкинга (разбиения) документов в RAG? Как размер чанка влияет на качество поиска?
13. Для чего используются векторные базы данных? Назовите известные векторные БД и их особенности.
14. Какие сценарии интеграции ИИ-моделей в корпоративные системы существуют? Опишите синхронный и асинхронный инференс.
15. Что такое Model Serving и какие существуют способы развертывания моделей (in-process, микросервис, serverless, dedicated serving)?
16. Какие требования предъявляются к API для ИИ-моделей (REST, gRPC, WebSocket)?
17. В чем заключается суть MLOps? Опишите компоненты CI/CD пайплайна для ML.
18. Какие инструменты используются для версионирования данных, кода и моделей?
19. Зачем нужна контейнеризация (Docker) в ML-проектах? Какую роль играет оркестрация (Kubernetes)?
20. Какие метрики необходимо отслеживать при мониторинге ИИ-системы в продуктивной среде?
21. Что такое дрейф данных (Data Drift) и дрейф концепта (Concept Drift)? Как их обнаружить и чем они различаются?
22. Какие уязвимости входят в OWASP Top 10 для LLM? Назовите основные угрозы.
23. Что такое промпт-инъекция (Prompt Injection)? Какие методы защиты от этой атаки существуют?
24. Что такое Red Teaming применительно к ИИ-безопасности и зачем оно проводится?
25. Раскройте концепцию Human-in-the-Loop (HITL). Приведите примеры степеней вовлечения человека.
26. Чем объяснимость модели (XAI) отличается от интерпретируемости? Какие методы XAI вы знаете?
27. Каковы основные принципы проектирования пользовательского опыта (UX) для генеративных ИИ-систем?
28. Как в RAG-системе можно визуализировать источники ответа для повышения доверия пользователя?
29. Как организовать цикл сбора обратной связи от пользователя для дообучения модели?
30. Какие этические аспекты необходимо учитывать при проектировании ИИ-систем (предвзятость, прозрачность, справедливость)?

Раздел 6. Оценочные средства промежуточной аттестации (с ключами)

1. Какая методология разработки наилучшим образом учитывает итеративный характер создания ИИ-систем с постоянным улучшением моделей?
А) Каскадная (Waterfall)
Б) Спиральная
В) V-образная
Г) CRISP-DM
Правильный ответ: Г (CRISP-DM)

2. Какой метод сжатия модели заключается в обучении меньшей модели имитировать выходы большей (учителя)?
А) Квантизация
Б) Прунинг
В) Дистилляция знаний

Г) Low-rank adaptation (LoRA)
Правильный ответ: В

3. Какой компонент RAG-пайплайна отвечает за поиск наиболее релевантных фрагментов документов?
А) LLM (Large Language Model)
Б) Векторная база данных и ретривер
В) Модуль аугментации данных
Г) Система мониторинга дрейфа
Правильный ответ: Б

4. Какая атака на LLM заключается во встраивании скрытых инструкций в текстовые данные, которые обрабатывает модель?
А) Data Poisoning
Б) Model Inversion
В) Indirect Prompt Injection
Г) Membership Inference
Правильный ответ: В

5. Что из перечисленного относится к задачам мониторинга (Observability) ИИ-системы в продуктивной среде?
А) Написание кода модели
Б) Разметка датасета
В) Детекция дрейфа данных (Data Drift)
Г) Выбор функции активации
Правильный ответ: В

6. Какие из перечисленных факторов являются ключевыми при выборе между использованием API закрытой модели и дообучением open-source модели?
А) Стоимость инференса при масштабировании
Б) Необходимость соблюдения требований о хранении данных на территории РФ
В) Цвет логотипа компании-разработчика
Г) Требования к латентности ответа
Правильный ответ: А, Б, Г

7. Какие методы относятся к параметро-эффективному дообучению (PEFT) больших языковых моделей?
А) LoRA (Low-Rank Adaptation)
Б) Полное дообучение всех весов
В) Prefix Tuning
Г) Дистилляция знаний
Правильный ответ: А, В

8. Какие инструменты используются для версионирования данных и управления пайплайнами в MLOps?
А) Git
Б) Docker
В) DVC (Data Version Control)
Г) MLflow
Правильный ответ: В, Г

9. Какие стратегии чанкинга (разбиения документов) применяются при построении RAG-систем?
А) Fixed-size chunking
Б) Semantic chunking

В) Recursive chunking
Г) Random chunking
Правильный ответ: А, Б, В

10. Какие метрики качества поиска (retrieval) используются для оценки RAG-пайплайна?
А) Hit Rate
Б) BLEU
В) Mean Reciprocal Rank (MRR)
Г) Perplexity
Правильный ответ: А, В

11. Установите правильную последовательность этапов жизненного цикла ИИ-системы:
А) Деплой и интеграция
Б) Сбор и разметка данных
В) Бизнес-анализ и определение метрик
Г) Мониторинг и переобучение
Д) Выбор архитектуры и обучение модели
Правильный ответ: В → Б → Д → А → Г

12. Установите последовательность этапов обработки запроса в RAG-системе:
А) Генерация ответа LLM на основе промпта и контекста
Б) Векторизация запроса с помощью эмбеддеров
В) Ранжирование и фильтрация результатов
Г) Поиск ближайших векторов в векторной БД
Д) Прием запроса от пользователя
Правильный ответ: Д → Б → Г → В → А

13. Установите последовательность шагов при дообучении (fine-tuning) модели с использованием LoRA:
А) Сохранение и регистрация адаптеров
Б) Заморозка весов базовой модели
В) Обучение малых матриц (адаптеров) на целевой задаче
Г) Загрузка базовой предобученной модели
Д) Добавление LoRA-слоев в архитектуру
Правильный ответ: Г → Б → Д → В → А

14. Установите последовательность шагов CI/CD пайплайна для ML-проекта:
А) Деплой модели на production
Б) Запуск автоматических тестов (юнит, интеграционных)
В) Валидация данных и моделей на staging
Г) Сборка Docker-образа
Д) Регистрация модели в Model Registry
Правильный ответ: Б → В → Д → Г → А

15. Установите последовательность действий при обработке промпт-инъекции в системе безопасности LLM:
А) Применение защитного системного промпта
Б) Фильтрация и санитизация ввода
В) Отклонение запроса или возврат безопасного ответа
Г) Детекция вредоносного паттерна (регулярками или моделью-модератором)
Правильный ответ: Б → А → Г → В

16. Установите соответствие между методом сжатия модели и его описанием:
Методы: (1) Квантизация, (2) Прунинг, (3) Дистилляция

Описания:
А) Удаление незначимых (близких к нулю) весов из сети
Б) Снижение точности представления весов (например, с FP32 до INT8)
В) Обучение малой модели имитировать выходы большой модели
Правильный ответ: 1-Б, 2-А, 3-В

17. Установите соответствие между компонентом RAG и его функцией:
Компоненты: (1) Эмбеддинг-модель, (2) Векторная БД, (3) Reranker
Функции:
А) Хранение и поиск векторных представлений
Б) Переранжирование найденных результатов для повышения релевантности
В) Преобразование текста в векторное представление
Правильный ответ: 1-В, 2-А, 3-Б

18. Установите соответствие между типом требований и примером:
Типы: (1) Функциональное требование, (2) Нефункциональное требование (атрибут качества), (3) Бизнес-требование
Примеры:
А) Система должна обрабатывать запрос пользователя менее чем за 500 мс
Б) Модель должна классифицировать отзывы на позитивные и негативные
В) Внедрение чат-бота должно снизить нагрузку на колл-центр на 30%
Правильный ответ: 1-Б, 2-А, 3-В

19. Установите соответствие между угрозой безопасности LLM и её описанием:
Угрозы: (1) Prompt Injection, (2) Data Poisoning, (3) Model Inversion
Описания:
А) Восстановление конфиденциальных данных из обученной модели
Б) Отравление обучающей выборки для внедрения вредоносного поведения
В) Игнорирование системных инструкций через специально сформированный запрос
Правильный ответ: 1-В, 2-Б, 3-А

20. Установите соответствие между инструментом и его назначением в MLOps:
Инструменты: (1) MLflow, (2) DVC, (3) Docker
Назначения:
А) Версионирование данных и управление пайплайнами
Б) Управление жизненным циклом моделей (логирование экспериментов, реестр моделей)
В) Контейнеризация приложений и воспроизводимость окружения
Правильный ответ: 1-Б, 2-А, 3-В

21. Утверждение: В RAG-системе увеличение размера чанка (фрагмента документа) всегда приводит к повышению точности ответа LLM, так как модель получает больше контекста.
Правильный ответ: Неверно. Увеличение чанка приводит к большему количеству шума и мусорного контекста, что может ухудшить качество ретривала и генерации.

22. Утверждение: Контейнеризация с помощью Docker обеспечивает полную воспроизводимость эксперимента, включая версии данных.
Правильный ответ: Неверно. Docker фиксирует окружение (библиотеки, ОС), но не данные. Для воспроизводимости данных нужны DVC или другие инструменты.

23. Утверждение: При использовании метода LoRA веса исходной большой модели замораживаются, а обучаются только малые матрицы-адаптеры.
Правильный ответ: Верно.

24. Утверждение: Приватность пользовательских данных (GDPR, ФЗ-152) не является проблемой при проектировании ИИ-систем, так как модели не хранят информацию о пользователях после обучения.
Правильный ответ: Неверно. Модели могут запоминать персональные данные, и к ним применимы атаки Model Inversion.

25. Утверждение: Ошибка на этапе сбора требований в ИИ-проекте обходится дешевле, чем ошибка на этапе развертывания модели в продакшене.
Правильный ответ: Верно. (Чем раньше ошибка обнаружена, тем дешевле её исправление).

Раздел 7. Перечень учебной литературы, необходимой для освоения дисциплины

7.1. Основная литература

1. Сотник С.Л. Проектирование систем искусственного интеллекта : учебное пособие. 2-е изд. Москва : Профобразование, 2024. 228 с. (Базовый учебник, охватывающий архитектуру систем ИИ, вопросы обучения, адаптации и проектирования интеллектуальных систем) .
2. Остроух А.В., Суркова Н.Е. Системы искусственного интеллекта : монография. 4-е изд., стер. Санкт-Петербург : Лань, 2024. 228 с. ISBN 978-5-507-47478-3. (Рассматриваются методы представления знаний и подходы к проектированию интеллектуальных систем, включая нейросетевые технологии) .
3. Ping D. The Machine Learning Solutions Architect Handbook : Design, Build, and Secure Scalable ML Systems. 2nd ed. Packt Publishing, 2024. 602 p. (Практическое руководство по проектированию ML-систем, MLOps, RAG и архитектурным паттернам для генеративного ИИ) .
4. Rothman D. RAG-Driven Generative AI : Build Enterprise-Ready GenAI Systems with Retrieval-Augmented Generation. 2nd ed. Packt Publishing, 2026. (Книга о проектировании RAG-систем, включая DualRAG, GraphRAG, мультиагентные системы и методы борьбы с галлюцинациями) .
5. Коллектив авторов. От чёрного ящика к инженерии : практическое руководство по проектированию LLM-систем. GitHub, 2026. Режим доступа: https://github.com/DenZNK/BlackboxBook (Открытая книга о LLMOps, RAG, промпт-инжиниринге, безопасности и наблюдаемости LLM-систем в production) .

7.2. Дополнительная литература

1. Kreuzberger D., et al. Machine Learning Operations (MLOps): Overview, Definition, and Architecture. 2023. (Академический обзор MLOps с детальным описанием принципов, компонентов и ролей) .
2. Zhang Z.Y. Foundations of GenAI Orchestration: RAG, MLOps, and LLMOps. Washington University in St. Louis, 2025. (Статья об оркестрации GenAI, включая эволюцию RAG от классического до агентного и Human-in-the-Loop) .
3. Сотник С.Л. Проектирование систем искусственного интеллекта : учебное пособие. Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. 228 с. ISBN 978-5-4497-0868-7. (Описывает архитектуру систем ИИ, распознавание образов, экспертные системы и методы анализа многомерных данных) .
4. Databricks. The Big Book of MLOps : Data-centric MLOps and LLMOps. TechTarget, 2026. (Практическое руководство по MLOps, управлению LLM, RAG и мониторингу через полный AI-жизненный цикл) .
5. Остроух А.В., Суркова Н.Е. Системы искусственного интеллекта: проектирование и разработка в транспортном комплексе. Санкт-Петербург : Лань, 2024. (Монография с фокусом на прикладное проектирование ИИ-систем для транспортной отрасли) .

7.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Официальная документация LangChain : фреймворк для оркестрации LLM и RAG-пайплайнов. Режим доступа: https://python.langchain.com/ (дата обращения: 12.05.2026).
2. Официальная документация MLflow : платформа для управления жизненным циклом ML-моделей. Режим доступа: https://mlflow.org/ (дата обращения: 12.05.2026).
3. GitHub - BlackboxBook: «От чёрного ящика к инженерии» — практическая книга об LLM-системах. Режим доступа: https://github.com/DenZNK/BlackboxBook (дата обращения: 12.05.2026).
4. Электронно-библиотечная система «Лань». Раздел «Искусственный интеллект. Проектирование систем ИИ». Режим доступа: https://e.lanbook.com (дата обращения: 12.05.2026).
5. TechTarget Hub - MLOps и LLMOps ресурсы. Режим доступа: https://www.techtarget.com/hub/MLOps (дата обращения: 12.05.2026).

Раздел 8. Материально-техническая база и информационные технологии

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине:

Материально-техническое обеспечение дисциплины **«Проектирование систем с использованием технологий искусственного интеллекта»** включает в себя учебные аудитории для проведения занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, самостоятельной работы обучающихся.

Учебные аудитории укомплектованы специализированной мебелью и техническими средствами обучения. Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети Интернет.

Дисциплина может реализовываться с применением дистанционных технологий обучения. Специфика реализации дисциплины с применением дистанционных технологий обучения устанавливается дополнением к рабочей программе. В части не противоречащей специфике, изложенной в дополнении к программе, применяется настоящая рабочая программа.

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине с применением дистанционных образовательных технологий включает в себя:

Компьютерная техника, расположенная в учебном корпусе Института (ул. Качинцев, 63, кабинет Центра дистанционного обучения):

1. Intel i 3 3.4Ghz\ОЗУ 4Gb\500GB\RadeonHD5450
2. Intel PENTIUM 2.9GHz\ОЗУ 4GB\500GB

3 личные электронные устройства (компьютеры, ноутбуки, планшеты и иное), а также средства связи преподавателей и студентов.

Информационные технологии, необходимые для осуществления образовательного процесса по дисциплине с применением дистанционных образовательных технологий включают в себя:

- система дистанционного обучения (СДО) (Learning Management System) (LMS) Moodle (Modular Object-Oriented Dynamic Learning Environment);
- электронная почта;
- система компьютерного тестирования;
- Цифровой образовательный ресурс IPR SMART;
- система интернет-связи skype;
- телефонная связь;
- ПО для организации конференций.

Обучение обучающихся инвалидов и обучающихся с ограниченными возможностями здоровья осуществляется посредством применения специальных технических средств в зависимости от вида нозологии.

При проведении учебных занятий по дисциплине используются мультимедийные комплексы, электронные учебники и учебные пособия, адаптированные к ограничениям здоровья обучающихся.

Лекционные аудитории оборудованы мультимедийными кафедрами, подключенными к звуковым колонкам, позволяющими усилить звук для категории слабослышащих обучающихся, а также проекционными экранами, которые увеличивают изображение в несколько раз и позволяют воспринимать учебную информацию обучающимся с нарушениями зрения.

При обучении лиц с нарушениями слуха используется усилитель слуха для слабослышащих людей Cyber Ear модель HAP-40, помогающий обучаемым лучше воспринимать учебную информацию.

Обучающиеся с ограниченными возможностями здоровья, обеспечены печатными и электронными образовательными ресурсами (программы, учебники, учебные пособия, материалы для самостоятельной работы и т.д.) в формах, адаптированных к ограничениям их здоровья и восприятия информации:

для лиц с нарушениями зрения:

- в форме электронного документа;
- в форме аудиофайла;

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.

Раздел 9. Методические указания для обучающихся по освоению дисциплины

Дисциплина включает практические занятия, самостоятельную работу обучающегося.

В ходе изучения дисциплины «**Проектирование систем с использованием технологий искусственного интеллекта**» перед обучающимися стоит задача не только закрепить знания о сложных информационных явлениях, о чем свидетельствует содержание тематического плана, глубоко разобраться в объемном учебном материале, но и сформировать у себя на основе полученных компьютерных знаний соответствующие профессионально важные качества.

Практические занятия – один из самых эффективных видов учебных занятий, на которых обучающиеся учатся творчески работать с различной информацией, являются также действенной формой активизации самостоятельной работы обучающихся.

Целью практических занятий является закрепление полученных в ходе лекций, а также в ходе самостоятельной работы над учебной и специальной литературой, знаний, умений и навыков. На практических занятиях особо обращается внимание на умение обучающихся проявлять элементы творчества в процессе самостоятельной работы, применять полученные знания на практике.

Практические занятия занимают центральное место в учебном процессе, так как позволяют на завершающем этапе усвоения материала, после прослушанной лекции и самостоятельного поиска дополнительных сведений по рассматриваемой проблематике, окончательно уточнить, сформировать свои позиции в ходе работы в составе учебной группы.

Основное в подготовке и проведении практикума – это самостоятельная работа обучающегося над изучением темы лекционного материала. Практические занятия проводятся по специальным планам – заданиям, которые содержатся в материалах, подготовленных на кафедре. Обучающийся обязан точно знать план занятия либо конкретное задание к нему.

При подготовке к практическим занятиям следует чаще обращаться к справочной литературе, полнее использовать консультации (групповые и индивидуальные, устные и письменные) с преподавателями, которые читают лекции и проводят практикумы.

Таким образом, в процессе подготовке к практическому занятию рекомендуется:

- ознакомиться с вопросами плана;
- прочитать конспект лекции по изучаемой теме;
- прочитать соответствующие главы учебников, статьи;
- просмотреть перечень научных источников, предлагаемых в рабочей программе, выбрав несколько из них для углубленного изучения данной темы.

По каждому практическому заданию обучающиеся отчитываются преподавателю, оформляя письменный отчёт, в котором сохраняют результаты своей работы в виде файлов. Результаты выполнения практических заданий оцениваются с учетом теоретических знаний по соответствующим вопросам дисциплины и уровнем владения практическими навыками при работе на компьютере.

Для углубленного изучения и освоения материала целесообразно выполнение практических работ, наряду с другими различными формами обучения обучающихся: тесты, задачи, упражнения, которые используются при проведении практических занятий, выполнении контрольных и аудиторных работ, а также при самостоятельном изучении данной дисциплины.

Одним из наиболее интенсивных способов изучения дисциплины является самостоятельное выполнение практических работ, на которых вырабатываются навыки построения web-сайтов.

СРО позволяет глубже освоить теоретические и практические вопросы, понять принципы программирования web-сайтов и научиться создавать свои интернет приложения.

Основными задачами организации процесса самостоятельной работы по дисциплине являются:

- приобретение знаний по теоретическим основам построения систем искусственного интеллекта, являющихся дополнением к материалу лекционных аудиторных занятий;
- приобретение практических навыков по созданию систем искусственного интеллекта.

Основные формы реализации СРО – изучение учебно-методической литературы по разработке систем искусственного интеллекта. В качестве базовой литературы можно использовать учебники и учебные пособия, согласно приведенному списку в разделе 6 рабочей программы, а также любые другие источники информации, такие как электронные учебники, обучающие и эн-

циклопедические сайты, публикации журналов и конференций.

Обучающийся допускается к зачетному занятию по результатам успешного выполнения всех практических заданий и самостоятельной работы.

Учебно-методическое издание

Рабочая программа учебной дисциплины

Проектирование систем с использованием технологий искусственного интеллекта

(Наименование дисциплины в соответствии с учебным планом)

Скоробогатченко Дмитрий Анатольевич

(Фамилия, Имя, Отчество составителя)