

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет компьютерных технологий и прикладной математики
Центр искусственного интеллекта

УТВЕРЖДАЮ:
Проректор по учебной работе, качеству образования – первый проректор
тор
_____ Хагуров Т.А.
подпись
«02» июня 2026



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.04 ВВЕДЕНИЕ В СПЕЦИАЛЬНОСТЬ

Направление подготовки 02.03.02 Фундаментальная информатика и информационные технологии

Направленность (профиль) Современные методы машинного обучения и компьютерного зрения

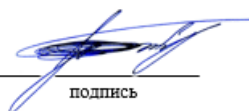
Форма обучения очная

Квалификация бакалавр

Краснодар 2026

Рабочая программа дисциплины Введение в специальность составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии

Программу составил(и):
А.В. Коваленко, проф., д. техн. .н.
И.О. Фамилия, должность, ученая степень, ученое звание


подпись

Рабочая программа дисциплины утверждена на заседании центра искусственного интеллекта
протокол № 04 «14» мая 2026 г.
Руководитель центра ИИ Коваленко А.В.


подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики
протокол № 03 «15» мая 2025 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 бюджетное образовательное учреждение 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 целостного представления о специальности, связанной с применением генеративных ИИ-моделей (LLM), освоение базовых компетенций в области промпт-инжиниринга, развитие навыков командной работы в гибридной среде «человек + ИИ» и понимания трендов современного искусственного интеллекта для постановки и решения профессиональных задач.

Ознакомление с практикой использования генеративного ИИ для быстрого прототипирования в реальном секторе экономики (на примере игровой индустрии) через выполнение кейсов индустриального партнёра (МИП), включая создание MVP игровых продуктов и обучение RL-агентов в симуляционной среде.

Изменения связанные с включение в рабочую программу дисциплины материалов индустриального партнера помечены аббревиатурой МИП

Аббревиатура МИП – «Материалы индустриального партнера»

1.2 Задачи дисциплины

1. Ознакомить студентов с историей развития, современными трендами и ключевыми направлениями искусственного интеллекта, включая генеративные модели (ChatGPT, DeepSeek, GigaChat, YandexGPT, Qwen, Claude, Gemini и др.).

2. Сформировать умение позиционировать собственную задачу в контексте трендов ИИ и выбирать перспективные методы её решения.

3. Научить базовым техникам промпт-инжиниринга (шаблоны, цепочки рассуждений, системные промпты) и анализу реакции моделей.

4. Развить навыки эффективной коммуникации в проектной команде, включая постановку задач людям и ИИ-агентам, фиксацию договорённостей и критериев качества.

5. Подготовить к участию в коллективной разработке ИИ-решений с использованием промптов и генеративных моделей.

6. Сформировать практические компетенции по применению генеративного ИИ для создания MVP игрового продукта (на основе кейсов индустриального партнёра МИП):

- разработка промптов для генерации игровых ассетов (текстуры, звуковое сопровождение, анимация, C#-скрипты);
- настройка пайплайна интеграции с игровыми движками (Unity);
- проектирование системных промптов для отбора и оценки сгенерированного контента;
- освоение базовых принципов обучения RL-агентов в физической симуляции (среда Unity ML-Agents, функция награды, наблюдения).

7. Познакомить студентов с ролями AI-инженера-прототипировщика и ML-инженера в игровой индустрии, а также с методами оценки эффективности (ROI) внедрения генеративного ИИ при разработке игровых прототипов.

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Введение в специальность» относится к «Части, формируемой участниками образовательных отношений» Блока 1 «Дисциплины (модули)» учебного плана.

Реализация дисциплины предполагает взаимодействие с индустриальным партнёром – студией CarX TECHNOLOGIES. В рамках дисциплины студенты выполняют практические кейсы (МИП), ориентированные на реальные задачи игровой индустрии, что способствует формированию профессиональных компетенций, востребованных на рынке труда.

Пререквизиты: Дисциплина реализуется на 1 курсе в первом семестре.

Смежные дисциплины: Разработка гибридных интеллектуальных систем (SS-2.1(C))

1.4 Профессиональные роли в структуре образовательной программы

Роль 1: Data Engineer (Инженер по данным)

Задачи:

1. Проектирование и построение ETL-процессов
2. Создание и оптимизация хранилищ данных
3. Обеспечение качества и доступности данных
4. Настройка инфраструктуры для обработки больших данных
5. Интеграция разрозненных источников данных
6. Работа с данными в области природопользования, медицины, связи и телекоммуникаций

Роль 2: ML Engineer (Инженер МО)

Задачи:

1. Реализация ML-моделей в продуктивных системах
2. Оптимизация производительности и масштабирование моделей
3. Разработка ML-пайплайнов и автоматизация процессов
4. Мониторинг качества моделей в продуктиве
5. Интеграция ML-решений с бизнес-приложениями

Роль 3: MLOps (Специалист по эксплуатации ИИ)

Задачи:

1. Автоматизация процессов обучения и развертывания моделей
2. Мониторинг производительности ML-систем
3. Управление версиями моделей и данных
4. Обеспечение CI/CD для ML-проектов
5. Оптимизация вычислительных ресурсов

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, уровень и формулировка компетенции	Индикаторы	Уровни освоения индикаторов компетенции
ЛС-8 Способен применять знания об истории развития и трендах современного ИИ для формулирования корректных постановок задач и поиска перспективных способов решения проблем с помощью ИИ	ЛС-8.1 Позиционирует собственную задачу в заданной области знания с точки зрения трендов современного ИИ	Знает: – основные этапы истории развития ИИ и генеративных моделей; – ключевые тренды современного ИИ (LLM, мультимодальность, RAG, агентные системы); – базовые направления ИИ (NLP, компьютерное зрение, анализ данных); – (МИП) особенности применения генеративного ИИ в игровой индустрии: создание MVP, генерация ассетов, обучение RL-агентов.
		Умеет: – формулировать задачу в своей предметной области; – соотносить поставленную задачу с актуальными технологиями ИИ; – указывать перспективные методы решения (zero-shot, few-shot, дообучение и др.);

		– (МИП) формулировать задачу по прототипированию игрового продукта и соотносить её с возможностями LLM и диффузионных моделей.
		Владеет: – навыком анализа трендов для обоснования выбора ИИ-решения; – способностью критически оценивать применимость генеративных моделей к конкретной задаче; – (МИП) навыком анализа применимости генеративных моделей для конкретных этапов разработки игры (текстуры, звук, код, поведение AI).
	LLM-5.1 Использует базовые шаблоны промптов	Знает: – концепцию промпта и его роль в диалоге с LLM; – базовые шаблоны: zero-shot, few-shot, chain-of-thought, role-based prompting; – структуру промпта (инструкция, контекст, входные данные, примеры)
		Умеет: – составлять промпты по заданному шаблону для типовых задач (генерация текста, ответ на вопрос, классификация); – адаптировать шаблоны под предметную область; – (МИП) составлять промпты для генерации C# скриптов в Unity (управление автомобилем, ИИ-преследование) с использованием шаблонов zero-shot и few-shot. – (МИП) генерировать описания текстур, звуковых дорожек и анимации через LLM для последующей передачи в специализированные модели (YandexART, диффузионные сети).
LLM-5 Организует взаимодействие с генеративными моделями через проектирование, анализ и применение промптов	LLM-5.6 Анализирует и отлаживает промпты	Владеет: – навыком проектирования промптов для решения прикладных задач; – (МИП) навыком проектирования промптов для создания игровых механик и логики противников в физической симуляции
		Знает: – основные параметры генерации (temperature, top-p, max_tokens) и их влияние на результат; – типичные ошибки в промптах (двусмысленность, недостаток контекста, противоречия); – подходы к A/B-тестированию промптов

		Умеет: – изменять параметры генерации и анализировать реакцию модели; – проводить сравнение нескольких вариантов промпта; – выявлять причины нежелательного поведения модели (галлюцинации, смещение, отказ от ответа); – (МИП) проводить отладку промптов при генерации игровых ассетов и C#-кода: выявлять причины неработоспособности скриптов, несоответствия стиля текстур, некорректного поведения RL-агента; – (МИП) применять A/B-тестирование промптов для выбора наилучших вариантов генерации игровых уровней или поведения NPC. Владеет: – методикой итеративной отладки промптов; – способностью документировать изменения и результаты экспериментов; – (МИП) методикой итеративной отладки промптов при создании MVP игрового продукта (от постановки задачи до получения рабочей демо-версии в Unity).
SS-2 Способен осуществлять свою трудовую деятельность с учётом необходимости эффективной коммуникации и взаимодействия в рамках коллективной проектной работы в сфере ИИ	SS-2.1 Эффективно коммуницирует с участниками проектной команды при планировании, реализации и анализе результатов работы в контексте гибридной команды «Человек+ИИ», включая постановку задач людям и ИИ-агентам, фиксацию договорённостей и критериев качества	Знает: – роли и зоны ответственности в команде ИИ-проекта; – принципы постановки задач для людей и для ИИ-агентов; – понятие критериев готовности (Definition of Done) для задач, связанных с ИИ; – (МИП) роли и зоны ответственности в команде разработки игр с использованием ИИ: AI-прототипировщик, ML-инженер, технический дизайнер, интегратор. Умеет: – формулировать задачи для членов команды и для ИИ-сервисов с использованием шаблонов промптов; – фиксировать договорённости и критерии качества в письменном виде; – участвовать в обсуждении архитектурных и аналитических решений; – (МИП) формулировать задачи для ИИ-агентов при генерации игрового контента (текстуры, код, анимация) с учётом требований к стилю и производительности.

		– (МИП) фиксировать критерии качества (DoD) для сгенерированных ассетов (точность соответствия промпту, частота ошибок в коде, визуальное сходство). Владеет: – навыками работы в гибридной команде (люди + ИИ); – способностью адаптировать свою коммуникацию под технический контекст задачи; – (МИП) навыками работы в гибридной команде с использованием инструментов Unity AI Assistant и генеративных моделей для ускорения прототипирования. – (МИП) способностью формулировать техническое задание для ИИ-генерации компонентов игрового проекта и проверять их на соответствие замыслу.
--	--	---

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

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

Вид учебной работы	Всего часов	Семестры (часы)
		1
Контактная работа, в том числе:	57,2	57,2
Аудиторные занятия (всего):	50	50
Занятия лекционного типа	16	16
Лабораторные занятия	34	34
Занятия семинарского типа (семинары, практические занятия)		
Иная контактная работа:	7,2	7,2
Контроль самостоятельной работы (КСР)	2	2
Промежуточная аттестация (ИКР)	5,2	5,2
Самостоятельная работа, в том числе:	14,8	14,8
Подготовка курсовой работы	5	5
Проработка учебного (теоретического) материала	5	5
Выполнение индивидуальных заданий (подготовка сообщений, презентаций)	4,8	4,8
Подготовка к текущему контролю		
Общая трудоемкость	час.	72
	в том числе контактная работа	57,2
	зач. ед	2

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

Разделы (темы) дисциплины, изучаемые в 1 семестре

№	Наименование разделов (тем)	Количество часов				
		Все го	Аудиторная ра- бота			Внеа- уди- тор- ная ра- бота
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
1.	Введение в генеративный ИИ и LLM	5	1		2	2
2.	Архитектуры генеративных моделей	7	2		4	1
3.	Принципы промпт-инжиниринга	7	2		4	1
4.	Шаблоны и стратегии промптов	7	2		4	1
5.	Создание пайплайнов взаимодействия с ИИ	7	2		4	1
6.	Интеграция промптов в бизнес-кейсы	5	-		4	1
7.	Отладка, тестирование и метрики качества	6	1		4	1
8.	Мультимодальные промпты и интерфейсы	6	1		4	1
9.	Этические и правовые аспекты ИИ.	4,8	1		2	1,8
10.	Игровой продукт как вид интерактивного приложения	4	2		-	2
11.	Прикладное использование генеративного ИИ для создания прототипа игрового про- дукта	6	2		2	2
ИТОГО по разделам дисциплины		64,8	16		34	14,8
Контроль самостоятельной работы (КСР)		2				
Промежуточная аттестация (ИКР)		5,2				
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 теку- щего кон- троля	Соот- вет- ствие индика- торам компе- тенций
1	2	3	4	5
1	Введение в генеративный ИИ и LLM	История развития и основные тренды современного ИИ от логических систем до больших языковых моделей. Понятие генеративного ИИ. Различие между дискриминативными и генеративными моделями. История и развитие LLM. Обзор ключевых применений (ассистенты, генерация текста, анализ данных, автоматизация).	Т	LC-8.1
2	Архитектуры генеративных моделей	Transformer, Attention, Positional Encoding – базовые механизмы. Сравнение ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen,	Т	LC-8.1

		Claude, Gemini, Manus, Diffusion-моделей. Архитектурные особенности LLM для текстов и моделей для изображений/звука. Обзор инструментов: Hugging Face, OpenAI API, Diffusers.		
3	Принципы промпт-инжиниринга	Что такое промпт и зачем он нужен. Промпт как программный интерфейс к модели. Контекст, токенизация, температурные параметры и параметры вывода. Основные ошибки в промптах и их влияние на генерацию.	T	LLM-5.1, LLM-5.6
4	Шаблоны и стратегии промптов	Zero-shot, One-shot, Few-shot, Chain-of-thought prompting. Шаблоны: Q&A, таблицы, списки, role-based prompting. Использование system prompts и инструкций. Интерактивные цепочки промптов.	T	LLM-5.1
5	Создание пайплайнов взаимодействия с ИИ	Сценарии автоматизации: от простых скриптов до web-приложений. Интеграция промптов в чат-ботов, дашборды и REST API. Использование Python (FastAPI), Node.js, Postman, cURL для тестирования. Пример пайплайна: ввод → промпт → вызов LLM → результат → логика обработки.	T	LLM-5.1, SS-2.1
6	Интеграция промптов в бизнес-кейсы	Цель: продемонстрировать прикладную значимость промпт-инжиниринга. Сценарии: генерация email-ответов, кратких новостей, отчётов по данным. Формализация задачи → построение промпта → оценка результата.	T	SS-2.1, LC-8.1
7	Отладка, тестирование и метрики качества	Методы оценки результатов генерации: BLEU, ROUGE, METEOR, Self-BLEU. A/B тестирование промптов. Анализ ошибок модели, диагностика неожиданных ответов. Версионирование промптов и контроль изменений.	T	LLM-5.6
8	Мультимодальные промпты и интерфейсы	Работа с изображениями, аудио, видео: от CLIP и DALL·E до Gemini. Комбинированные промпты: «нарисуй + опиши», «расскажи о фото». Примеры использования: генерация инфографики, дизайн, synthetic data. Инструменты: Stable Diffusion, Midjourney, Whisper, Gemini.	T	LLM-5.1
9	Этические и правовые аспекты ИИ	Правовые и этические вызовы ИИ. Предвзятость (bias) в генеративных моделях. Приватность и защита данных. Ответственный ИИ. Защита от генеративных рисков. Профессиональная этика специалиста по ИИ.	T	LLM-5.6, LC-8.1
10	(МИП) Игровой продукт как вид интерактивного приложения	Базовые принципы создания игр: вид цифрового культурно-развлекательного продукта. Виды игр, монетизация, экономика, обучение, развитие и формирование качеств у пользователя (игрока). Технологии создания игр (платформы iOS, Android, PC), игровые редакторы	T	LC-8.1 (МИП), SS-2.1

		(Unity, Unreal, Unigine и др.). Роли и направления игровой индустрии как будущей профессиональной сферы.		
11	(МИП) Прикладное использование генеративного ИИ для создания прототипа игрового продукта	Понятие MVP (minimal viable product) и как его получить с помощью генеративного ИИ. Определение целостности прототипа и составление концепции с помощью ИИ. Инструментарий: Unity AI Assistant, MS Copilot, ChatGPT, Gemini и другие для создания кода, контента, текста и видеоряда игрового продукта. Генерация изображений, анимации, игровых механик с помощью ИИ.	Т	LC-8.1 (МИП), LLM-5.1, SS-2.1

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

Не предусмотрены

2.3.3 Лабораторные занятия

№	Наименование раздела (темы)	Наименование лабораторных работ Задание	Форма текущего контроля	Соответствие индикаторам компетенций
1	2	3	4	5
1	Введение в генеративный ИИ и LLM	Регистрация и запуск ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus, ввод различных вопросов: факты, мнения, креативные задачи. Сравнение ответов при одинаковом запросе, повторяемость. Анализ структуры ответа. Пример: задать один и тот же вопрос в стиле «студента», «профессора», «менеджера».	ЛР	LC-8.1, LLM-5.1
2	Архитектуры генеративных моделей	Сравнение архитектур ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus. Подключение к Hugging Face и запуск модели через Spaces. Анализ числа параметров, скорости, качества генерации. Исследование конфигурации модели. Использование интерфейса Inference API.	ЛР	LC-8.1
3	Принципы промпт-инжиниринга	Цель: Освоить работу с параметрами генерации и структурой промптов. Содержание: Температура, Top-p, Max tokens, Presence/penalty. Проведение серии запросов с изменением параметров и анализом результатов. Формулировка промптов в виде инструкций, вопросов, ролевых моделей.	ЛР	LLM-5.1, LLM-5.6

4	Шаблоны и стратегии промптов	Сравнение эффективности Zero-shot vs Few-shot. Построение цепочки размышлений в задачах логики (напр. задачи на монеты, путешествия). Создание промптов с примерами.	ЛР	LLM-5.1
5	Создание пайплайнов взаимодействия с ИИ	Установка FastAPI, создание эндпоинта POST /generate. Обработка входного текста, вызов модели (через OpenAI API или Hugging Face). Форматирование вывода. Тестирование с Postman или curl.	ЛР	LLM-5.1, SS-2.1
6	Интеграция промптов в бизнес-кейсы	Цель: Демонстрировать прикладную значимость промпт-инжиниринга. Сценарии: генерация email-ответов, кратких новостей, отчётов по данным. Формализация задачи → построение промпта → оценка результата.	ЛР	LLM-5.1, SS-2.1
7	Отладка, тестирование и метрики качества	Проведение серии генераций, подсчёт BLEU/ROUGE (с использованием готовых библиотек). Анализ точности и вариативности. Проведение А/В тестирования: два промпта – один набор входных данных. Создание таблицы с метриками и субъективной оценкой.	ЛР	LLM-5.6
8	Мультимодальные промпты и интерфейсы	Работа с Stable Diffusion (через web-интерфейс или Colab). Создание описания → генерация изображения. Работа с обратной задачей: CLIP/BLIP – генерация подписи к изображению.	ЛР	LLM-5.1
9	Этические и правовые аспекты ИИ	Генерация спорных текстов (фактчекинг, токсичность, дискриминация). Примеры использования фильтров (moderation API). Создание промпта для этичного взаимодействия. Анализ потенциальных нарушений.	ЛР	LLM-5.6, LC-8.1
10	(МИП) Прикладное использование генеративного ИИ для создания прототипа игрового продукта	Практика генеративного ИИ в создании прототипа игрового проекта. Составление и описание MVP игрового проекта, внешний тест идеи с помощью ChatGPT, DeepSeek, GigaChat, YandexGPT. Создание тестов, изображений, видеоряда, анимации для презентации идеи игры с помощью инструментов генеративного ИИ. • Внутренний тест идеи. Создание игровой механики проекта в редакторе Unity с помощью инструмента Unity AI Assistant. Получение демо-версии игровой механики.	ЛР	LC-8.1 (МИП), LLM-5.1, LLM-5.6, SS-2.1

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)

1. Разработка промптов для генерации Python/C-кода на основе текстового описания задачи

2. Сравнение качества генерации программ с помощью различных промптов (zero-shot vs few-shot)
3. Генерация кода циклов и условий: промпты для типовых задач (факториал, Фибоначчи и т.д.)
4. Создание промптов для автокомментирования кода на Python/C
5. Создание ассистента на базе LLM для решения задач по основам программирования
6. Анализ промптов для генерации функций сортировки и обработки массивов
7. Построение цепочек промптов для отладки кода и поиска ошибок
8. Автоматическая генерация тестов к простым Python/C -функциям с помощью LLM
9. Промпты для генерации кода с различной степенью детализации и пояснений
10. Сравнение выходов от ChatGPT и GitHub Copilot на одинаковые промпты по Python/C
11. Разработка промптов для вычисления пределов и производных с пояснением шагов
12. Промпты для генерации решений задач по интегралам (неопределённые и определённые)
13. Промпты, обучающие LLM решать уравнения и неравенства (включая пошаговые рассуждения)
14. Сравнение точности и корректности вывода решений по математическому анализу
15. Разработка промптов для визуализации графиков функций
16. Анализ ошибок при генерации решений по математическому анализу: примеры и способы повышения точности
17. Использование промптов для генерации кратких справок и теоретических сводок по темам
18. Построение промптов, объясняющих суть пределов, производных и их применений
19. Создание промптов-репетиторов: моделирование учебного диалога по математическому анализу
20. Интеграция LLM в Jupyter-ноутбуки для помощи в решении задач по математическому анализу
21. Генерация промптов для нахождения скалярного и векторного произведения
22. Построение промптов для расчета длины, нормализации и угла между векторами
23. Разработка промптов для решений задач с матрицами: транспонирование, умножение
24. Промпты для объяснения линейной зависимости и базиса
25. Моделирование диалога между студентом и LLM по решению СЛАУ
26. Использование промптов для визуализации векторов и линейных преобразований
27. Создание обучающих карточек по темам векторной алгебры с помощью LLM
28. Разработка тестов по темам линейной алгебры через генерацию LLM
29. Промпты для составления пошаговых решений по методу Гаусса
30. Сравнение промптов для символического и численного решения задач
31. Создание учебного бота для 1 курса: вопросы по Python /C, математическому анализу и векторной алгебре
32. Исследование эффективности генерации задач с решениями для подготовки к зачету
33. Сценарии применения промптов в электронных учебниках и курсах
34. Генерация задач олимпиадного уровня и пошаговых решений к ним
35. Автоматическое составление шпаргалки по темам курса
36. Промпты для генерации викторин и мини-игр по математике и программированию

37. Создание LLM-ассистента, который «объясняет, как преподаватель» — стилизация ответа
38. Разработка промптов для генерации простых визуальных задач по аналитической геометрии
39. Исследование возможности генерации контрпримеров и логических ловушек в задачах
40. Разработка генератора практикумов для подготовки к экзамену на базе промптов
41. Разработка промптов для автоматического анализа пользовательских отзывов
42. Построение промптов для генерации запросов на естественном языке
43. Сравнительный анализ качества аналитических отчетов, созданных с помощью LLM
44. Использование промптов для обобщения больших текстовых массивов в аналитике
45. Автоматическая визуализация данных по текстовому описанию с использованием LLM
46. Промпт-интерфейсы для объяснения метрик и KPI
47. Создание ассистента для анализа отчетности и подсветки ключевых трендов
48. Разработка шаблонов промптов для интерпретации результатов А/В-тестов
49. Генерация рекомендаций по улучшению метрик продукта на основе текстового отчета
50. Сравнение подходов chain-of-thought и few-shot prompting в аналитике данных
51. Встраивание LLM в пайплайн мониторинга моделей машинного обучения
52. Промпты для генерации документации ML-проекта по коду и логам
53. Промпты для генерации тестов и проверок качества моделей
54. Автоматическая генерация DAG-пайплайна для ML-задачи по описанию
55. Разработка системы версионирования и контроля промптов в продуктивной среде
56. Использование LLM для анализа логов и диагностики инцидентов
57. Создание инструментов отладки и наблюдаемости через генеративные интерфейсы
58. Генерация REST API документации по описанию модели
59. Построение ассистента для настройки гиперпараметров моделей
60. Интеграция LLM в CI/CD пайплайн проекта машинного обучения
61. Промпт-интерфейс для составления технического задания на ИИ-систему
62. Автоматизация составления project charter с помощью LLM
63. Промпт-интерфейсы для анализа рисков ИИ-проекта
64. Разработка шаблонов генерации ROI-оценок и бизнес-кейсов
65. Внедрение LLM-ассистента в трекинг задач и задач планирования спринтов
66. Использование LLM для анализа обратной связи по результатам ИИ-внедрения
67. Построение system prompt'ов для сценариев взаимодействия с заказчиком
68. Генерация текстов презентаций и отчетов по проекту ИИ
69. Исследование эффективности LLM в роли фасилитатора проектных совещаний
70. Создание мета-промптов для быстрой адаптации под задачи клиента
71. Промпты для генерации UX-интерфейсов по текстовому описанию
72. Разработка промптов для генерации изображений в корпоративной стилистике
73. Мультимодальные промпты: генерация инфографики по аналитическому описанию
74. Промпт-интерфейсы для создания сценариев обучения персонала
75. Генерация юридических документов на основе текстовых описаний требований
76. Разработка промптов для создания диалогов и FAQ-систем
77. Промпт-интерфейс для автоматической генерации вакансий и резюме

78. Промпты для генерации стратегий продвижения продуктов на основе описания ниши
79. Исследование надёжности и повторяемости вывода при генерации технических текстов
80. Этические фильтры и их влияние на точность генерации: отбор и тестирование промптов
81. Прототипирование MVP гоночной аркады с использованием генеративного ИИ (по кейсу 5 МИП).
82. Обучение RL агента для задачи «догони лидера» в физической симуляции (по кейсу 6 МИП).
83. Разработка системы генерации игровых уровней с помощью LLM и диффузионных моделей (МИП).
84. Создание игрового ассистента на базе LLM для динамической адаптации сложности (МИП)

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 29.08.2025
2	Решение задач	Методические указания по организации самостоятельной работы студентов, утвержденные УМК ФКТиПМ, протокол №1 от 29.08.2025

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 вышеназванных 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 «Введение в специальность».

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

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 зачете;

– 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	Код и наименование индикатора (в соответствии с п. 1.5)	Результаты обучения (в соответствии с п. 1.5)	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	LC-8.1 – Позиционирует собственную задачу в заданной области знания с точки зрения трендов современного ИИ	Знает: основные этапы истории развития ИИ и генеративных моделей; ключевые тренды (LLM, мультимодальность, RAG, агентные системы); базовые направления ИИ. Умеет: формулировать задачу, соотносить с направлениями ИИ, указывать актуальные технологии. Владеет: анализом трендов для выбора ИИ-решения, критической оценкой применимости генеративных моделей.	Тесты по истории и трендам ИИ; дискуссии «Какой ИИ подходит для задачи?»; кейс-анализ; тест из 20 вопросов (вопросы №1,2,9,15 относятся к данному индикатору); дискуссия по кейсам МИП (MVP, RL-агент); анализ эффективности ИИ в игровом прототипировании	Вопросы к зачёту №1–10, 15–17; №41–45 (МИП); практическое задание №2 (сравнение эффективности шаблонов); №21 (генерация C# скрипта), №23 (функция награды)
2	LLM-5.1 – Использует базовые шаблоны промптов	Знает: концепцию промпта, базовые шаблоны (zero-shot, few-shot, chain-of-thought, role-based), структуру промпта. Умеет: составлять промпт по шаблону, адаптировать под предметную область. Владеет: навыком проектирования промптов для прикладных задач.	Лабораторные работы №3, 4; отчёты по шаблону; мини-тест на знание шаблонов; лабораторная работа №10 (МИП) – генерация игровых ассетов и скриптов; тест (№3,4,10,13,18); практикум по составлению промптов для Unity	Вопросы к зачёту №11–13, 18; практические задания №1 (автокомментирование), №4 (генерация email), №5 (LLM для Python), №8 (сценарии голосового ассистента); №21 (C# скрипт), №22 (системный промпт для текста), №24 (генерация описания и промпта для изображения)
3	LLM-5.6 – Анализирует и отлаживает промпты	Знает: параметры генерации (temperature, top-p, max_tokens), типичные ошибки в промптах, подходы к A/B-тестированию. Умеет: изменять параметры, анализировать реакцию модели, проводить сравнение, выявлять причины нежелательного поведения. Владеет: методикой итеративной отладки, документированием экспериментов.	Лабораторная работа №7 (A/B-тестирование, анализ ошибок); отчёты по отладке; таблицы сравнения; лабораторная работа №10 (отладка промптов для генерации игрового кода); тест (№5,6,11,16); анализ галлюцинаций при генерации скриптов	Вопросы к зачёту №14, 19; практические задания №6 (тестирование параметров), №14 (сравнение двух промптов), №17 (чек-лист); №23 (функция награды – отладка параметров обучения), №25 (пайплайн интеграции LLM в Unity)

4	SS-2.1 – Эффективно коммуницирует с участниками проектной команды в контексте гибридной команды «Человек+ИИ»	Знает: роли в команде ИИ-проекта, принципы постановки задач для людей и ИИ-агентов, Definition of Done (DoD). Умеет: формулировать задачи для членов команды и ИИ-сервисов, фиксировать договорённости и критерии качества, участвовать в обсуждении. Владеет: навыками работы в гибридной команде, способностью формулировать запросы к ИИ и фиксировать DoD.	Ролевая игра (лабораторная работа №5); групповое обсуждение критериев качества; фиксация DoD в мини-проектах; лабораторная работа №10 (работа в команде над игровым прототипом, распределение задач между людьми и ИИ); дискуссия по кейсам МИП (роль AI-прототиповщика, критерии приёмки ассетов)	Вопросы к зачёту №20–25; практические задания №3 (конструктор резюме), №9 (промпт для генерации SQL), №18 (анализ шаблонов); №22 (системный промпт для отбора текстов), №25 (описание пайплайна с указанием ролей и DoD)
---	--	---	--	--

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

Пример теста для проведения текущего контроля знаний

Вопрос 1

Какой этап истории ИИ связан с появлением генеративных моделей типа GPT и диффузионных сетей?

- А) Логический подход (1950–1970)
- В) Экспертные системы (1980)
- С) Эпоха больших языковых моделей (2020-е)
- D) Первая зима ИИ (1974–1980)

Правильный ответ: С

Вопрос 2

Что из перечисленного является примером тренда современного ИИ, согласно рабочей программе?

- А) Использование магнитных лент для хранения данных
- В) Мультимодальность (одновременная работа с текстом, изображениями, аудио)
- С) Полная замена программистов нейросетями
- D) Отказ от нейронных сетей в пользу символьных методов

Правильный ответ: В

Вопрос 3

Какой шаблон промпта предполагает предоставление модели одного примера решения задачи перед основным запросом?

- А) Zero-shot
- В) One-shot
- С) Few-shot
- D) Chain-of-thought

Правильный ответ: В

Вопрос 4

Какая задача НЕ относится к типичному использованию генеративного ИИ при создании MVP игрового продукта (по кейсу CarX)?

- А) Генерация C# скрипта управления автомобилем

- В) Создание текстур дорожного покрытия через YandexART
- С) Компиляция бинарного кода движка Unity
- D) Генерация описаний трёх фоновых треков

Правильный ответ: С

Вопрос 5

Параметр temperature в генеративных моделях отвечает за:

- А) Длину выходного текста
- В) Степень случайности / креативности ответа
- С) Количество повторяющихся слов
- D) Язык ответа

Правильный ответ: В

Вопрос 6

В ходе отладки промпта для генерации C# скрипта в Unity вы заметили, что скрипт не компилируется из-за синтаксических ошибок. Какое действие поможет исправить ситуацию?

- А) Увеличить max_tokens
- В) Добавить в промпт фразу «Напиши корректный C# код для Unity» и пример правильного синтаксиса
- С) Уменьшить top_p до 0.1
- D) Переключить модель с GPT-4 на GPT-3.5

Правильный ответ: В

Вопрос 7

Что означает Definition of Done (DoD) при постановке задачи ИИ-агенту в гибридной команде?

- А) Максимальное количество токенов в ответе
- В) Чётко сформулированные критерии качества и завершённости результата
- С) Срок, к которому задача должна быть выполнена
- D) Имя агента в системе

Правильный ответ: В

Вопрос 8

Какая роль в команде разработки игр с использованием ИИ отвечает за настройку пайплайна генерации контента и сборку прототипа в Unity?

- А) Game Designer
- В) AI Prototyper (инженер-прототипировщик ИИ)
- С) 2D Artist
- D) QA Engineer

Правильный ответ: В

Вопрос 9

Что такое MVP (Minimum Viable Product) в контексте разработки игры с помощью генеративного ИИ?

- А) Финальная версия игры со всеми уровнями
- В) Минимально работающий продукт, который можно показать фокус-группе для проверки концепции
- С) Техническая документация по движку
- D) Маркетинговый план релиза

Правильный ответ: В

Вопрос 10

ТОМ

Chain-of-thought prompting – это:

- А) Обучение модели на цепочке последовательных примеров
- В) Промпт, побуждающий модель показать пошаговые рассуждения перед ответом
- С) Замена слов в запросе синонимами
- D) Использование системного сообщения для задания роли

Правильный ответ: В

Вопрос 11

Какое явление LLM называется «галлюцинацией» (hallucination)?

- А) Отказ модели отвечать на вопрос
- В) Генерация моделью правдоподобных, но фактически неверных утверждений
- С) Повторение одного и того же слова несколько раз
- D) Ответ на языке, отличном от языка запроса

Правильный ответ: В

Вопрос 12

При обучении RL-агента в Unity ML-Agents функция награды (reward) служит для:

- А) Генерации текстур дороги
- В) Оценки качества сгенерированного кода
- С) Указания агенту, какие действия являются полезными или вредными
- D) Задания максимальной длины эпизода

Правильный ответ: С

Вопрос 13

Какой инструмент, упомянутый в кейсах МИП, позволяет генерировать C# скрипты внутри редактора Unity на основе промптов?

- А) Copilot
- В) Unity AI Assistant
- С) Midjourney
- D) Whisper

Правильный ответ: В

Вопрос 14

При разработке MVP гоночной аркады с помощью ИИ, какой компонент, скорее всего, придётся писать вручную, а не генерировать?

- А) Базовую физику столкновений (сложная, требующая точной настройки)
- В) Текстуры асфальта
- С) Рекламный текст для презентации
- D) Описание звукового сопровождения

Правильный ответ: А

Вопрос 15

Что из перечисленного НЕ является базовым направлением искусственного интеллекта согласно РПД?

- А) NLP (обработка естественного языка)
- В) Компьютерное зрение
- С) Квантовые вычисления
- D) Анализ данных

Правильный ответ: С

Вопрос 16 – множественный выбор

Какие параметры генерации можно настраивать при вызове LLM через API для управления характером ответа? (Выберите три верных)

- A) temperature
- B) font_size
- C) top_p
- D) max_tokens
- E) background_color

Правильные ответы: A, C, D

Вопрос 17 – множественный выбор

Какие из перечисленных задач могут быть решены с помощью генеративного ИИ при создании игрового прототипа (по кейсу CarX)? (Выберите все верные)

- A) Генерация трёхмерной сетки сложного персонажа (high-poly)
- B) Генерация текстовых описаний для внутриигровых событий
- C) Генерация C# скриптов для управления ИИ-противника
- D) Создание фоновой музыки в стиле дрифт через описание для аудио-модели
- E) Компиляция сборки Unity под Android

Правильные ответы: B, C, D

Вопрос 18 – множественный выбор

Какие типы промптов рекомендуется использовать для генерации игрового C# скрипта в Unity? (Выберите два)

- A) Промпт с ролью («Ты – архитектор кода в Unity»)
- B) Промпт без указания языка программирования
- C) Промпт с примером (few-shot) ожидаемого кода
- D) Промпт, содержащий только вопрос «Как управлять машиной?» без дополнительных указаний

Правильные ответы: A, C

Вопрос 19 (Соответствие)

Установите соответствие между инструментом / методом и его назначением в рамках дисциплины (в том числе кейсов МИП). Напишите пары буква-цифра.

Инструмент / метод	Назначение
1. Unity AI Assistant	A. Генерация изображений по текстовому описанию
2. Chain-of-thought	B. Подсказка модели пошагово рассуждать перед ответом
3. YandexART	C. Генерация C# скриптов внутри игрового движка
4. Hugging Face Spaces	D. Платформа для демонстрации и запуска моделей без установки

Правильные ответы:

- 1 – C
- 2 – B
- 3 – A
- 4 – D

Вопрос 20 (Краткий ответ)

Опишите (2–3 предложения), какие критерии качества (DoD) вы бы установили для задачи «Сгенерировать промпт для получения текстуры дорожного покрытия в стиле «ночной город с мокрым асфальтом» через диффузионную модель».

Пример правильного ответа (но возможны вариации):

DoD включает: 1) текстура имеет разрешение не менее 1024×1024; 2) визуально соответствует стилю «ночной город, мокрый асфальт» (наличие отражений, тёплого освещения); 3) отсутствуют видимые артефакты генерации (нарушенная геометрия,

пятна); 4) промт воспроизводим и даёт сходный результат при повторном запуске с тем же seed.

Таблица соответствия вопросов индикаторам

№ вопроса	Индикатор
1	LC-8.1
2	LC-8.1
3	LLM-5.1
4	LLM-5.1 + МИП
5	LLM-5.6
6	LLM-5.6 + МИП
7	SS-2.1
8	SS-2.1 + МИП
9	LC-8.1 + МИП
10	LLM-5.1
11	LLM-5.6
12	LC-8.1 + МИП
13	LLM-5.1 + МИП
14	SS-2.1 + МИП
15	LC-8.1
16	LLM-5.6
17	SS-2.1 + МИП
18	LLM-5.1 + МИП
19	LC-8.1 / LLM-5.1 / МИП
20	SS-2.1 + МИП

Практические кейсы по тематике лабораторных работ

Лабораторная работа №1: Введение в генеративный ИИ и LLM

Кейс: «Ассистент-справочник для студента 1 курса»

Задача: Сформулируйте 5 различных промтов, которые могли бы использовать первокурсники для генерации справок по основам программирования, математике и векторной алгебре.

Что нужно сделать:

Провести генерацию через ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus;

Оценить релевантность и полноту ответов;

Сделать выводы о влиянии формулировки промта.

Инструменты: ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus (markdown-отчёт), MS Word или Jupyter

Лабораторная работа №2: Архитектуры генеративных моделей

Кейс: «Выбор модели для создания образовательного ассистента»

Задача: Сравнить архитектуры GPT-2, GPT-3.5, GPT-4, DeepSeek, Perplexity, Grok и BERT с точки зрения применимости для генерации учебных материалов.

Что нужно сделать:

Изучить документацию Hugging Face;

Подключить модели через интерфейс или API;

Провести генерацию по одному и тому же сценарию;

Сравнить результаты по качеству и скорости.

Инструменты: Hugging Face, OpenAI Playground, Jupyter

Лабораторная работа №3: Принципы промпт-инжиниринга

Кейс: «Ответ в стиле преподавателя»

Задача: Разработать серию промптов, нацеленных на генерацию ответов в разных стилях (официальный, дружелюбный, краткий).

Что нужно сделать:

Сравнить результат генерации при использовании разных system prompt'ов;

Выделить ключевые фразы, влияющие на стиль;

Подготовить сравнительный анализ.

Инструменты: OpenAI Playground, DeepSeek, ChatGPT, Perplexity, Grok таблица Excel или Jupyter

Лабораторная работа №4: Шаблоны и стратегии промптов

Кейс: «Генерация ответа на вопрос по математическому анализу»

Задача: Решить задачу на определённый интеграл через zero-shot, few-shot и chain-of-thought prompting.

Что нужно сделать:

Написать по 3 промпта для одной задачи;

Провести генерацию;

Сравнить ход рассуждений модели.

Инструменты: ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus, Google Docs (отчёт), Jupyter

Лабораторная работа №5: Создание пайплайнов взаимодействия с ИИ

Кейс: «Мини-сервис по генерации текстов»

Задача: Реализовать веб-интерфейс, который отправляет текстовый запрос к модели и выводит результат.

Что нужно сделать:

Настроить FastAPI эндпоинт;

Организовать отправку запроса к OpenAI API;

Протестировать ответы модели через curl/Postman.

Инструменты: Python, FastAPI, OpenAI API, curl/Postman

Лабораторная работа №6: Интеграция промптов в бизнес-кейсы

Кейс: «Генератор email для отдела продаж»

Задача: Разработать серию промптов, автоматизирующих подготовку писем клиентам.

Что нужно сделать:

Выделить типовые шаблоны писем;

Составить промпты под разные ситуации (приветствие, напоминание, предложение);

Провести тестирование на релевантность.

Инструменты: ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus, таблица Excel, Markdown-документ

Лабораторная работа №7: Отладка, тестирование и метрики качества

Кейс: «Промпты для подготовки к зачету»

Задача: Оценить эффективность двух разных промптов, создающих объяснение темы по векторной алгебре.

Что нужно сделать:

Сформировать два промпта;

Оценить результат с точки зрения полноты, точности, ясности;

Рассчитать BLEU или ROUGE (с помощью библиотеки).

Инструменты: ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus, nltk/rouge-score, Jupyter

Лабораторная работа №8: Мультимодальные промпты и интерфейсы

Кейс: «Обложка для онлайн-курса»

Задача: Сгенерировать изображение и сопроводительный текст для онлайн-курса по программированию.

Что нужно сделать:

Написать промпт для Stable Diffusion (через web-интерфейс);

Сформулировать title и описание для страницы курса;

Проверить соответствие изображения и текста.

Инструменты: Stable Diffusion WebUI / Leonardo.ai / DALL·E, DeepSeek, ChatGPT, Perplexity, Grok

Лабораторная работа №9: Этические аспекты и защита от генеративных рисков

Кейс: «Безопасный учебный чат-бот»

Задача: Проанализировать поведение модели при провокационных запросах. Разработать корректные формулировки, минимизирующие риски.

Что нужно сделать:

Сформулировать 5 спорных/пограничных запросов;

Оценить поведение модели;

Предложить промпты для безопасной генерации.

Инструменты: ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus, OpenAI Moderation API (по желанию), Markdown-отчёт

Лабораторная работа №10: (МИП) Прикладное использование генеративного ИИ для создания прототипа игрового продукта

Кейс: (МИП). Прототипирование MVP гоночной аркады с использованием генеративного ИИ

Контекст. Студия CarX TECHNOLOGIES хочет быстро проверить новую концепцию уличной гоночной аркады с элементами тюнинга. Задача: за 2 недели создать минимально жизнеспособный продукт (MVP), который можно показать фокус-группе. В команде нет художников и звукорежиссёров – только разработчики, владеющие промпт-инжинирингом и интеграцией ИИ. Разрешено использовать любые генеративные модели для создания ассетов и фрагментов кода.

Роль студента. AI-инженер-прототипировщик, отвечающий за настройку пайплайна генерации контента и сборку прототипа в Unity.

Задание (кейс).

1. Разработайте промпты для генерации: (а) трёх фоновых треков (дрифт, стрит-рейсинг, ночной город) через LLM (описание для аудио-модели); (б) набора текстур дорожного покрытия и авто через YandexART; (в) базовых скриптов управления автомобилем и ИИ противника на промпте «создай C# скрипт для следования по трассе с поворотами».

2. Опишите архитектуру сборки MVP: какие компоненты генерируются ИИ, какие пишутся вручную.

3. Предложите критерии оценки качества сгенерированного контента (например, соответствие стилю, частота ошибок в коде).
4. Напишите системный промпт для агента, который помогает отбирать лучшие варианты текстур из 10 сгенерированных.
5. Оцените временные затраты: сколько человеко-часов сэкономило использование ИИ по сравнению с традиционной разработкой.

Ожидаемый результат:

- дорожная карта создания MVP с указанием used AI tools;
- примеры промптов и сгенерированных результатов (скриншоты, фрагменты кода);
- оценка эффективности (ROI) применения ИИ в прототипировании.

Кейс: (МИП). Обучение RL-агента для задачи «догони лидера» в физической симуляции

Контекст. В игре CarX Street игроки жалуются на слишком предсказуемых противников. Нужен адаптивный AI-соперник, который учится преследовать лидера гонки, используя реальную физику автомобиля (инерция, сцепление с дорогой). Команда уже имеет симуляционную среду на Unity ML-Agents. Требуется обучить агента, который может догонять и удерживаться за целевым автомобилем на трассе с поворотами, не вылетая.

Роль студента. ML-инженер, настраивающий среду, функцию награды и параметры обучения PPO.

Задание (кейс).

1. Опишите пространство наблюдений (observations) для агента: какие параметры (относительная скорость, угол, расстояние, текущая скорость, показания датчиков касания дороги) необходимо передавать.
2. Предложите функцию награды (reward function) на каждом шаге, которая поощряет: сокращение дистанции до лидера, нахождение в зоне «след в след» (tailgating), прохождение контрольных точек; и штрафует: вылет с трассы, столкновение, слишком большое отставание.
3. Какие действия (discrete/continuous) должны быть у агента: газ/тормоз/руль? Обоснуйте.
4. Опишите конвейер обучения: как собирать опыт, как часто обновлять политику, как избегать переобучения под конкретную трассу.
5. Предложите метод оценки качества обученного агента (например, среднее время удержания лидера, количество обгонов за 5 кругов).
6. Напишите псевдокод цикла сбора данных для обучения с подкреплением (agent step → environment step → reward → buffer).

Ожидаемый результат:

- спецификация среды и RL-параметров;
- графики обучения (expected reward per episode);
- видеодемонстрация поведения обученного агента (описание того, что должно быть видно);
- рекомендации по развёртыванию модели в рантайме игры.

Оба кейса могут быть выбраны студентом для выполнения лабораторной работы №10 (МИП) или в качестве основы курсового проекта (темы 81, 82).

Пример лабораторной работы

Лабораторная работа №3: Принципы промпт-инжиниринга

Название: Влияние параметров генерации и структуры промпта на результат

Цель:

Научиться проектировать эффективные промпты и понимать влияние параметров генерации на выход LLM.

Задачи:

1. Сравнить генерацию текста при использовании разных структур промпта (вопрос, инструкция, ролевая модель).
2. Изменить параметры генерации (temperature, top_p, max_tokens) и проанализировать результат.
3. Сделать выводы о принципах точного управления генерацией.

Ожидаемые результаты:

- Умение формировать промпты разных типов.
- Понимание ключевых параметров генерации.
- Навык анализа влияния параметров на результат.

Инструменты и библиотеки:

- ChatGPT, DeepSeek, Perplexity, Grok, Llama, GigaChat, YandexGPT, Qwen, Claude, Gemini, Manus / OpenAI Playground
- Python (опционально, через OpenAI API)
- Jupyter Notebook / Google Colab

Исходные данные: Примеры тем: «Объясни, что такое производная», «Напиши письмо преподавателю», «Составь список рекомендаций»

Ход работы:

1. Выберите 3 темы (см. выше).
2. Для каждой темы сформулируйте по 3 промпта:
 - В виде вопроса
 - В виде инструкции
 - В виде диалога (роль: преподаватель)
3. Сгенерируйте текст при следующих параметрах:
 - Температура: 0.2, 0.7, 1.0
 - Top_p: 1.0, 0.9
 - Max tokens: 50, 100
4. Оформите таблицу с результатами.
5. Проанализируйте:
 - Какие типы промптов дают наиболее точный результат?
 - Как влияет температура на вариативность?
 - Как изменяется стиль и длина при изменении max_tokens?

Пример промпта:

Ты — преподаватель по математике. Объясни студенту, что такое производная, простым языком.

Пример кода (опционально):

```
import openai
openai.api_key = 'your-key'
response = openai.ChatCompletion.create(
    model="gpt-3.5-turbo",
    messages=[{"role": "user", "content": "Объясни, что такое производная"}],
    temperature=0.7,
    top_p=1.0,
    max_tokens=100
)
print(response['choices'][0]['message']['content'])
```

Анализ результатов:

- Температура 0.2 даёт шаблонный, формальный ответ.
- Температура 1.0 — более креативный и иногда некорректный результат.
- Ролевая модель даёт более понятный и дружелюбный ответ.

Требования к отчёту:

- Таблица с параметрами и результатами генерации
- Примеры промптов (минимум 3 темы × 3 структуры)
- Аналитическая часть с выводами
- Скриншоты или вывод кода

Критерии оценки (зачтено/незачтено):

Зачтено: все типы промптов реализованы, минимум 6 генераций, таблица оформлена, выводы осмысленны.

Незачтено: отсутствуют 2+ варианта промпта, нет анализа, неполный отчёт.

Лабораторная работа №5: Создание пайплайнов взаимодействия с ИИ

Название: Разработка API-интерфейса генерации текста по промпту

Цель:

Реализовать простой сервис, отправляющий запрос к LLM через OpenAI API и возвращающий результат.

Задачи:

1. Настроить FastAPI-приложение с POST-запросом.
2. Отправить промпт на генерацию в модель OpenAI.
3. Обработать и вернуть ответ пользователю.

Ожидаемые результаты:

- Умение использовать API для генерации текста
- Навык работы с FastAPI и REST-интерфейсами
- Опыт логирования и форматирования вывода

Инструменты и библиотеки:

- Python 3.10+
- FastAPI
- OpenAI API
- Uvicorn, requests, dotenv

Ход работы:

1. Создайте файл .env с вашим API-ключом:

OPENAI_API_KEY=sk-...

2. Создайте файл main.py:

```
from fastapi import FastAPI, Request
import openai, os
from dotenv import load_dotenv
load_dotenv()
```

```
openai.api_key = os.getenv("OPENAI_API_KEY")
```

```
app = FastAPI()
```

```
@app.post("/generate")
```

```
async def generate(request: Request):
```

```
    data = await request.json()
```

```
    prompt = data.get("prompt")
```

```
    response = openai.Completion.create(
```

```
        engine="text-davinci-003",
```

```
        prompt=prompt,
```

```
        temperature=0.7,
```

```
        max_tokens=150
```

```
    )
```

```
    return {"result": response.choices[0].text.strip() }
```

3. Запустите сервер:

```
uvicorn main:app --reload
```

4. Протестируйте через Postman или curl:

```
curl -X POST http://127.0.0.1:8000/generate -H "Content-Type:
```

```
application/json" -d '{"prompt": "Напиши поздравление с Днем программиста"}'
```

Требования к отчёту: Скриншоты кода, Логика работы запроса, Примеры промптов и ответов, Описание структуры API и параметров

Критерии оценки: Зачтено: API запущен, протестирован, вывод получен, структура кода и отчета корректна. Незачтено: ошибки в коде, нет запуска, промпт не работает.

Зачетно-экзаменационные материалы для промежуточной аттестации (зачет)

Вопросы для подготовки к зачету

1. Что такое генеративный искусственный интеллект? Примеры его применения.
2. Как соотносятся задачи Data Science и задачи генеративного ИИ?
3. Объясните архитектуру трансформеров. Почему они эффективны для LLM?
4. Чем отличаются GPT и BERT по архитектуре и назначению?
5. Что такое Diffusion-модели и где они применяются?
6. Как работает температурный параметр в генерации текста?
7. Что такое top-p sampling и зачем он используется?
8. Какие параметры влияют на стиль и точность ответа LLM?
9. Почему важно учитывать контекст в промпт-инжиниринге?
10. Что такое zero-shot и когда он применяется?
11. Приведите пример few-shot промпта и опишите его преимущества.
12. В чём заключается метод Chain-of-Thought?
13. Объясните разницу между system prompt и user prompt.
14. Что такое пайплайн взаимодействия с LLM? Приведите пример.
15. Какие языки программирования используются для интеграции LLM?
16. Как можно реализовать генерацию текста через REST API?
17. Какие сложности возникают при встраивании промптов в пайплайн?
18. Какие бизнес-задачи можно автоматизировать с помощью LLM?
19. Приведите примеры шаблонов промптов для деловой переписки.
20. Как оценить ROI от внедрения генеративных моделей?
21. Что включает в себя формализация бизнес-задачи для ИИ?
22. Как можно измерить качество ответа LLM?
23. В чём смысл метрик BLEU, ROUGE и METEOR?
24. Что такое A/B тестирование промптов?
25. Как повысить надёжность генерации в нестабильной среде?
26. Что такое мультимодальный промпт?
27. Как работает генерация изображений по описанию?
28. Какие инструменты используются для мультимодальной генерации?
29. Какие проблемы возникают при обработке изображений и текста вместе?
30. Какие риски связаны с генерацией фейков и токсичных ответов?
31. Какие существуют методы фильтрации недопустимого контента?
32. Как учитывать юридические и этические нормы при работе с LLM?
33. Что такое ответственное ИИ и как оно связано с промпт-инжинирингом?
34. Какие этапы включает разработка эффективного промпта?
35. Как формализовать требования к проекту генеративного ассистента?
36. В чём отличие линейной и итеративной логики промпта?
37. Как документировать и версионировать промпты?
38. Какие навыки необходимы промпт-инженеру?
39. Какие инструменты облегчают работу с промптами?
40. Какие критерии используются для защиты курсовой работы по промпт-инжинирингу?
41. Опишите понятие MVP (minimum viable product) в контексте игрового проекта. Как генеративный ИИ может ускорить создание MVP? (МИП)

42. Какие компоненты игрового прототипа (код, текстуры, звук, анимация) можно генерировать с помощью LLM и диффузионных моделей? Приведите примеры промптов. (МИП)
43. Что такое Unity AI Assistant? Как он интегрируется с генеративными моделями для создания C# скриптов и игровых механик? (МИП)
44. Опишите основные элементы пространства наблюдений (observations) и функции награды (reward) для RL-агента, обучающегося преследовать лидера в гоночной игре. (МИП)
45. Как оценить ROI (экономическую эффективность) применения генеративного ИИ при прототипировании игрового продукта? Какие метрики следует учитывать? (МИП)

Практические задания к зачету

1. Промпты для автокомментирования кода
Задание: Напишите промпт, который получает функцию на Python и возвращает её подробный построчный комментарий. Протестируйте минимум 3 функции разной сложности.
2. Сравнение эффектов шаблонов генерации
Задание: Сравните качество ответов LLM на один и тот же вопрос в zero-shot, few-shot и chain-of-thought вариантах. Оцените точность, полноту и интерпретируемость.
3. Конструктор резюме
Задание: Реализуйте промпт-интерфейс, который получает ключевые данные о студенте (имя, опыт, образование) и генерирует готовое резюме. Вариативность: 2 языка и 3 стилистики (деловой, творческий, краткий).
4. Бизнес-кейс: генерация email
Задание: Сформируйте шаблон промпта, который получает информацию о клиенте и цели письма, и возвращает вежливое деловое письмо. Предусмотрите управление длиной и тоном.
5. LLM для обучения Python
Задание: Разработайте серию промптов-объяснений (не менее 5) для начинающих программистов. Темы: циклы, условия, функции, списки, рекурсия.
6. Тестирование параметров генерации
Задание: Один и тот же промпт выполните с разными параметрами temperature и top_p. Сравните результаты по разнообразию и адекватности вывода.
7. Мультимодальный генератор баннеров
Задание: Используя Stable Diffusion или DALL·E, создайте промпты для генерации рекламных баннеров по текстовому описанию продукта. Протестируйте минимум 3 случая.
8. Сценарии голосового ассистента
Задание: Напишите промпты, моделирующие диалоги голосового помощника в разных ролях: репетитор, медик, оператор доставки. Структурируйте промпты и ответы.
9. Промпт-интерфейс для генерации SQL
Задание: Составьте промпт, который преобразует текстовое описание запроса (например, «выбери все имена, где возраст больше 30») в корректный SQL-код. Протестируйте 5 кейсов.
10. Автоматический генератор викторин
Задание: Создайте промпт, который получает тему и количество вопросов, и генерирует викторину с 4 вариантами ответов и ключами. Оформите в формате JSON.
11. API-интерфейс генерации технических описаний
Задание: Напишите REST API на FastAPI, который получает название ИТ-продукта и генерирует маркетинговое или техническое описание через OpenAI API.
12. Реформаулировка и упрощение текста
Задание: Разработайте промпт, который получает научный абзац и переформулирует его «на пальцах» для школьника. Примеры: производная, логарифм, ядро матрицы.

13. Метаметамет: генератор метаметов
Задание: Напишите метамет, который по краткому описанию задачи (например: «нужно сгенерировать SEO-заголовок») создаёт эффективный метамет для LLM.
14. Отладка и сравнение метаметов
Задание: Сравните два метмета, решающих одну задачу (например, генерация обобщения статьи). Проведите анализ метрик качества (BLEU, ROUGE) и субъективной оценки.
15. Формализация этики взаимодействия
Задание: Составьте список тем, потенциально нарушающих правила безопасности (hate speech, вредные советы и т.д.). Разработайте безопасные альтернативные метметы.
16. Контекстный чат-бот по дисциплине
Задание: Сформируйте 10 метметов, покрывающих ключевые темы дисциплины (вопросы, определения, сравнения), и объедините их в логичный диалог с LLM.
17. Модель оценки эффективности метметов
Задание: Разработайте чек-лист оценки метметов (по критериям: краткость, точность, адаптивность, управляемость). Примените к 5 вашим метметам.
18. Анализ шаблонов генерации задач
Задание: Составьте метмет, который получает тему (например, производная) и генерирует учебную задачу с решением. Проведите оценку валидности и сложности.
19. Версионирование метметов
Задание: Реализуйте в Git или другой системе хранения файл с историей изменения метметов, комментариями и результатами генерации. Отрадите влияние изменений.
20. Курсовой ассистент: чат по теме дисциплины
Задание: Создайте интерактивную систему (на основе Gradio, Streamlit или Telegram-бота), в которой метметы получают контекст из введённых пользователем данных. Поддержите темы Python, линейной алгебры и анализа.
21. (МИП) Разработка метмета для генерации C# скрипта управления автомобилем в Unity
Задание: Напишите метмет (на русском или английском языке), который при отправке в ChatGPT (или DeepSeek, GigaChat) сгенерирует C# скрипт для управления автомобилем в гоночной игре (газ, тормоз, поворот). Укажите, какие параметры генерации (temperature, max_tokens) вы выберете. Проверьте работоспособность сгенерированного кода в Unity (или опишите ожидаемый результат).
22. (МИП) Системный метмет для отбора лучших текстур
Задание: Предложите системный метмет для LLM, которая будет анализировать 10 вариантов текстур дорожного покрытия (сгенерированных YandexART / Stable Diffusion) и выбирать 3 лучших по критериям: реалистичность, соответствие стилю «ночной город», отсутствие артефактов. Приведите пример ожидаемого вывода.
23. (МИП) Функция награды для RL-агента «догони лидера»
Задание: Напишите псевдокод функции награды (reward function) для обучения агента в среде Unity ML-Agents. Агент должен получать поощрение за сокращение дистанции до лидера и прохождение контрольных точек, а штраф – за вылет с трассы и столкновение. Объясните, почему вы выбрали именно такие коэффициенты.
24. (МИП) Генерация игрового описания и визуального метмета
Задание: Сформулируйте текстовое описание (концепт) уличной гоночной аркады с элементами тюнинга. На основе этого описания создайте метмет для генерации:
 - рекламного текста (через LLM);
 - изображения автомобиля в стиле киберпанк (через диффузионную модель).Представьте полученные результаты (реальные или ожидаемые).
25. (МИП) Пайплайн интеграции LLM в Unity
Задание: Опишите архитектуру пайплайна, в котором запрос из игры (например, «сгенерируй уникальное описание для победителя гонки») отправляется к LLM через

REST API, а ответ отображается в интерфейсе. Укажите, какие шаги необходимы для реализации в Unity (корутины, парсинг JSON, обработка ошибок). Приложите фрагмент C# кода для отправки HTTP-запроса к API GigaChat или ChatGPT.

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 зачет. Студенты обязаны сдать зачет в соответствии с расписанием и учебным планом.

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

Зачет 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 зачета: устно.

Результат сдачи зачета заносится преподавателем в зачетную ведомость и зачетную книжку.

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

Критерии оценки:

1. Оценка ответов на вопросы к зачету (50% итоговой оценки)

Зачет

Полные, развернутые ответы с демонстрацией глубокого понимания темы.

Ответы содержат основные идеи, но без углубленного анализа.

Использование примеров, формул, корректных терминов.

Возможны небольшие ошибки в деталях или формулировках.

Умение анализировать и сравнивать методы.

% выполнения: 60–100% (допускаются незначительные неточности).

Незачет

Отсутствие понимания ключевых концепций.

Грубые ошибки или неспособность ответить на большую часть вопросов.

% выполнения: <60%.

2. Оценка выполнения практических кейсов и лабораторных работ (50% итоговой оценки)

Зачет

Полное выполнение всех этапов кейса с инновационными решениями.

Достижение целевых метрик (например, $F1 > 0.9$).

Четкая документация кода и анализ результатов.

% выполнения: 60–100%.

или

Выполнены основные задачи, но без дополнительной оптимизации.

Незначительные отклонения от целевых метрик (например, $F1 = 0.85$).

или

Решены базовые задачи, но с критическими ошибками.

Низкое качество кода или отсутствие анализа.

Незачет

Невыполнение ключевых этапов.

Код нерабочий или отсутствует.

% выполнения: <60%.

Методические рекомендации, определяющие процедуры оценивания лабораторных работ:

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 зачете;

– 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.

План-проспект методических указаний по организации лабораторных работ по дисциплине "Введение в специальность"

1. Общие сведения

Образовательная программа: Искусственный интеллект и аналитика данных

Дисциплина: " Введение в специальность "

Вид обеспечения: Проведение лабораторных работ

Условия применения:

- Курс рассчитан на студентов 1-го года обучения.
- Наличие доступа к вычислительным ресурсам (GPU/CPU).
- Использование открытых датасетов и библиотек, доступ к БЯМ.

2. Цели, задачи и ожидаемые результаты

Цели:

- Закрепление теоретических знаний на практике.
- Развитие навыков работы с Промпт запросами к БЯМ в профессиональной деятельности.

- Подготовка к решению реальных задач в индустрии.

Задачи:

- Обеспечить студентов структурированными лабораторными работами.
- Предоставить доступ к необходимым вычислительным ресурсам.
- Организовать проверку и обратную связь по выполненным работам.

Ожидаемые результаты:

- Умение применять Промпт запросы к БЯМ на практике.
- Навыки работы с базовым техникам промт-инжиниринга: шаблоны, цепочки рассуждений, system prompts.
- Опыт создания собственных промтов и анализу их эффективности.

3. Порядок реализации

3.1. Задача №1: Подготовка лабораторных работ

1) Определение тем:

- Введение в генеративный ИИ и LLM
- Архитектуры генеративных моделей
- Принципы промт-инжиниринга
- Шаблоны и стратегии промтов
- Создание пайплайнов взаимодействия с ИИ
- Интеграция промтов в бизнес-кейсы
- Отладка, тестирование и метрики качества
- Мультимодальные промты и интерфейсы
- Этические аспекты и защита от генеративных рисков
- Курсовая работа и защита

2) Разработка заданий:

- Пошаговые инструкции.
- Примеры кода.
- Контрольные вопросы.

Разработка заданий для лабораторной работы «Принципы промт-инжиниринга».

Название: Влияние параметров генерации и структуры промпта на результат

Цель: Научиться проектировать эффективные промты и понимать влияние параметров генерации на выход LLM.

Задачи:

1. Сравнить генерацию текста при использовании разных структур промпта (вопрос, инструкция, ролевая модель).
2. Изменить параметры генерации (temperature, top_p, max_tokens) и проанализировать результат.
3. Сделать выводы о принципах точного управления генерацией.

Ожидаемые результаты:

- Умение формировать промты разных типов.
- Понимание ключевых параметров генерации.
- Навык анализа влияния параметров на результат.

Инструменты и библиотеки:

- DeepSeek, ChatGPT, Perplexity, Grok / OpenAI Playground
- Python (опционально, через OpenAI API)
- Jupyter Notebook / Google Colab

Исходные данные: Примеры тем: «Объясни, что такое производная», «Напиши письмо преподавателю», «Составь список рекомендаций»

Ход работы:

1. Выберите 3 темы (см. выше).
2. Для каждой темы сформулируйте по 3 промпта:
 - В виде вопроса
 - В виде инструкции
 - В виде диалога (роль: преподаватель)
3. Сгенерируйте текст при следующих параметрах:
 - Температура: 0.2, 0.7, 1.0
 - Top_p: 1.0, 0.9
 - Max tokens: 50, 100
4. Оформите таблицу с результатами.
5. Проанализируйте:
 - Какие типы промптов дают наиболее точный результат?
 - Как влияет температура на вариативность?
 - Как изменяется стиль и длина при изменении max_tokens?

Пример промпта:

Ты — преподаватель по математике. Объясни студенту, что такое производная, простым языком.

Пример кода (опционально):

```
import openai
openai.api_key = 'your-key'
response = openai.ChatCompletion.create(
    model="gpt-3.5-turbo",
    messages=[{"role": "user", "content": "Объясни, что такое производная"}],
    temperature=0.7,
    top_p=1.0,
    max_tokens=100
)
print(response['choices'][0]['message']['content'])
```

Анализ результатов:

- Температура 0.2 даёт шаблонный, формальный ответ.
- Температура 1.0 — более креативный и иногда некорректный результат.
- Ролевая модель даёт более понятный и дружелюбный ответ.

Требования к отчёту:

- Таблица с параметрами и результатами генерации
- Примеры промптов (минимум 3 темы × 3 структуры)
- Аналитическая часть с выводами
- Скриншоты или вывод кода

Критерии оценки (зачтено/незачтено):

Зачтено: все типы промптов реализованы, минимум 6 генераций, таблица оформлена, выводы осмысленны.

Незачтено: отсутствуют 2+ варианта промпта, нет анализа, неполный отчёт.

Формируемые компетенции:

- ЛС 8 Б Способен применять знания об истории развития и трендах современного ИИ для формулирования корректных постановок задач и поиска перспективных способов решения проблем с помощью ИИ

– LLM-5 Б Организует взаимодействие с генеративными моделями через про-
ектирование, анализ и применение промптов

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

5.1 Основная литература:

1. Чернышев, С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2022. — 286 с. — (Высшее образование). — ISBN 978-5-534-14350-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/496893>. (дата обращения: 19.07.2025).
2. Платонов, А. В. Машинное обучение : учебное пособие для вузов / А. В. Платонов. — Москва : Издательство Юрайт, 2022. — 85 с. — (Высшее образование). — ISBN 978-5-534-15561-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/508804> (дата обращения: 19.07.2025).
3. Рабчевский, А. Н. Синтетические данные и развитие нейросетевых технологий : учебное пособие для вузов / А. Н. Рабчевский. — Москва : Издательство Юрайт, 2024. — 187 с. — (Высшее образование). — ISBN 978-5-534-17716-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/545036> (дата обращения: 19.07.2025).
4. Елисеев А. И., Минин Ю. В. Разработка программных интерфейсов веб-приложений с использованием фреймворка FastAPI : учебное пособие. Тамбов: ТГТУ, 2024. 81 с. <https://e.lanbook.com/book/472310> (дата обращения: 19.07.2025).
5. Лиманова Н. И. Разработка интеллектуальных чат-ботов : учебное пособие. Самара: ПГУТИ, 2024. <https://e.lanbook.com/book/463568> (дата обращения: 19.07.2025).

5.2 Дополнительная литература:

1. Ф. Джеймс, Т. Майк Промт-инжиниринг для GenAI. Паттерны надежных запросов для качественных результатов, Sprint Book, 2025, 432 с.
2. А.А. Костин Промпт-инжиниринг. Язык будущего. 2025. Ridero, 594 с.
3. В.А. Петров, А.Н. Тихонов. Искусственный интеллект и обработка естественного языка.
4. Златопольский Д. М. Основы программирования на языке Python. 2-е изд. Москва: ДМК Пресс, 2018.
5. А.А. Кузнецов. Машинное обучение для обработки естественного языка.
6. Руководство Google по промпт-инжинирингу. Часть 1: основы и базовые техники <https://habr.com/ru/articles/901426/>
7. Документация облачных сервисов cloud.ru, yandex.cloud.
8. <https://education.yandex.ru/handbook/prompting>
9. <https://yandex.cloud/ru/training/training-pro#Data>
10. Elshall AS and Badir A (2025) Balancing AI-assisted learning and traditional assessment: the FACT assessment in environmental data science education. Front. Educ. 10:1596462. doi: 10.3389/educ.2025.1596462 (Статья о рисках снижения базовых навыков при чрезмерной опоре на ИИ).
11. Wharton Knowledge “Without Guardrails, Generative AI Can Harm Education” (2024) <https://knowledge.wharton.upenn.edu/article/without-guardrails-generative-ai-can-harm->

[education/#:~:text=practice%20session%2C%20yet%20scored%20about,exam%20as%20the%20control%20group](#) Статья– эксперимент о влиянии GPT на обучение: зависимость от ИИ снижает глубокое усвоение материала.

12. Kadurin, Artur, et al. "The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology." *Oncotarget* 8.7 (2016): 10883.
13. 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.
14. Polykovskiy, Daniil, et al. "Molecular sets (MOSES): a benchmarking platform for molecular generation models." *Frontiers in pharmacology* 11 (2020): 565644.
15. Khrabrov, Kuzma, et al. " ∇^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.
16. Polykovskiy, Daniil, et al. "Entangled conditional adversarial autoencoder for de novo drug discovery." *Molecular pharmaceutics* 15.10 (2018): 4398-4405.
17. Николенко, Сергей, Кадури, Артур и Архангельская Екатерина. Глубокое обучение. "Издательский дом" Питер", 2017.
18. <https://openreview.net/forum?id=FMMF1a9ifL>
19. <https://openreview.net/forum?id=ElUrNM9U8c#discussion>
20. <https://openreview.net/forum?id=JoO6mtCLHD>
21. <https://aclanthology.org/2024.findings-emnlp.760/>
22. <https://aclanthology.org/2020.coling-main.588/>
23. https://link.springer.com/chapter/10.1007/978-3-030-72113-8_30
24. https://link.springer.com/chapter/10.1007/978-3-031-42448-9_10
25. <https://aclanthology.org/2024.findings-naacl.288/>
26. 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
27. 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
28. 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>,

5.3. Периодические издания:

1. Базы данных компании «Ист Вью» <http://dlib.eastview.com>
2. Электронная библиотека GREBENNIKON.RU <https://grebennikon.ru/>

5.4. 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. Президентская библиотека им. Б.Н. Ельцина <https://www.prilib.ru/>
8. База данных CSD Кембриджского центра кристаллографических данных (CCDC) <https://www.ccdc.cam.ac.uk/structures/>
9. Springer Journals: <https://link.springer.com/>
10. Springer Journals Archive: <https://link.springer.com/>
11. Nature Journals: <https://www.nature.com/>
12. Springer Nature Protocols and Methods:
13. <https://experiments.springernature.com/sources/springer-protocols>
14. Springer Materials: <http://materials.springer.com/>
15. Nano Database: <https://nano.nature.com/>
16. Springer eBooks (i.e. 2020 eBook collections): <https://link.springer.com/>
17. "Лекториум ТВ" <http://www.lektorium.tv/>
18. Университетская информационная система РОССИЯ <http://uisrussia.msu.ru>

Бесплатные образовательные ресурсы

1. Jupyter Notebook – интерактивные вычисления
2. Visual Studio Code – редактор кода с поддержкой Python
3. Google Scholar/arXiv – доступ к научным публикациям

Ресурсы свободного доступа

1. Промпт и Промпт инжиниринг https://courses.sberuniversity.ru/generative_art/img/13
2. Principled Instructions Are All You Need for Questioning LLaMA-1/2, GPT-3.5/4 (<https://arxiv.org/abs/2312.16171>).
3. Nathan Hunter. The Art of Prompt Engineering with ChatGPT
4. ChatGPT Prompt Engineering for Developers <https://www.promptingguide.ai/>
5. КиберЛенинка <http://cyberleninka.ru/>;
6. Американская патентная база данных <http://www.uspto.gov/patft/>
7. Министерство науки и высшего образования Российской Федерации <https://www.minobrnauki.gov.ru/>;
8. Федеральный портал "Российское образование" <http://www.edu.ru/>;
9. Информационная система "Единое окно доступа к образовательным ресурсам" <http://window.edu.ru/>;
10. Единая коллекция цифровых образовательных ресурсов <http://school-collection.edu.ru/> .
11. Проект Государственного института русского языка имени А.С. Пушкина "Образование на русском" <https://pushkininstitute.ru/>;
12. Справочно-информационный портал "Русский язык" <http://gramota.ru/>;
13. Служба тематических толковых словарей <http://www.glossary.ru/>;
14. Словари и энциклопедии <http://dic.academic.ru/>;
15. Образовательный портал "Учеба" <http://www.ucheba.com/>;
16. 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 дисциплине «Введение в специальность» предусмотрено проведение лекционных занятий, на которых даётся систематизированное представление о генеративных моделях искусственного интеллекта (ИИ) и их применении в профессиональной деятельности. В ходе лекций студенты знакомятся с базовыми концепциями генеративного ИИ (LLM, diffusion-модели), принципами их архитектуры (Transformer, GPT, BERT, T5), методами взаимодействия с такими моделями через текстовые запросы (prompt engineering). Особое внимание уделяется анализу стратегий промпт-инжиниринга: zero-shot, few-shot, chain-of-thought prompting, а также современным практикам создания эффективных запросов к LLM. Демонстрируются практические примеры использования генеративных моделей в бизнесе, образовании, программировании, креативных индустриях. Обсуждаются ключевые риски генерации: галлюцинации, этические ограничения, защита персональных данных. После каждой лекции студентам рекомендуется выполнить прикладные задания — анализировать ответы LLM, экспериментировать с параметрами генерации и шаблонами промптов.

Лабораторные занятия посвящены формированию практических навыков работы с генеративными ИИ-моделями. Студенты последовательно осваивают методы промпт-инжиниринга на платформе ChatGPT, а также через API-интерфейсы (OpenAI, Hugging Face, YandexGPT и др.). Выполняются упражнения по настройке параметров генерации (temperature, top-p, max tokens), составлению шаблонов промптов для типовых задач (написание писем, объяснение кода, генерация описаний). Рассматриваются веб-технологии интеграции генеративных моделей: создание сервисов на FastAPI, запуск интерфейсов через Gradio или Streamlit. В ходе лабораторных занятий студенты также изучают методы тестирования и отладки промптов, проводят А/В-тестирование, оценивают качество генерации по метрикам (BLEU, ROUGE). Выполняются проекты по генерации текста, изображений и мультимодальных ответов. После каждого занятия выдаются задания для самостоятельной доработки — настройка собственных промптов, реализация цепочек взаимодействия с ИИ, анализ ответов модели.

Самостоятельная работа обучающихся направлена на углублённое изучение методологических и прикладных аспектов промпт-инжиниринга. Рекомендуется регулярно обращаться к учебникам, документации API и статьям по теме (включая публикации на arXiv, OpenAI Research, Google DeepMind). Студенты должны научиться формулировать прикладные задачи (автоматизация переписки, поддержка пользователей, генерация кода), разрабатывать промпты и пайплайны генерации, настраивать параметры вывода и оценивать релевантность результатов. Особое внимание уделяется разработке комплексных решений — от проектирования промптов до внедрения в виде web-интерфейса или CLI-инструмента. Умения в области Python-программирования и баз данных активно применяются при обработке входных/выходных данных и построении пользовательских интерфейсов.

Итоговой формой освоения курса является проектная (курсовая) работа. В её рамках студент разрабатывает полнофункциональный AI-инструмент, использующий промпты и генеративные модели. Это может быть чат-бот, генератор маркетинговых текстов, ассистент преподавателя, визуализатор знаний и др. Курсовой проект охватывает все ключевые элементы: анализ предметной области, построение промптов, тестирование и отладка, реализация пайплайна взаимодействия, анализ рисков и ограничений. Такой подход позволяет студенту проявить творческую инициативу и продемонстрировать уровень сформированных компетенций.

Для студентов с ограниченными возможностями здоровья предусмотрены индивидуальные консультации и адаптированные материалы. Преподаватель помогает осваивать интерфейсы взаимодействия с ИИ, объясняет ключевые понятия в доступной форме, предоставляет инструкции с альтернативным форматированием. При необходимости используются голосовые интерфейсы, увеличенный масштаб экрана, сопровождение при выполнении заданий. Индивидуальный подход обеспечивает равные условия участия в образовательном процессе и достижения запланированных результатов обучения.

Рассмотрим примеры кейсов.

Подход, определяющий установление соответствия кейсов ИП и УГТ (5), позволяет четко соотносить этапы развития технологии с вовлеченностью партнера и снижать риски при переходе от лабораторных испытаний к промышленному внедрению.

УГТ5 - Проверка данного уровня проводится в средах имитационного моделирования эмулирующих условия приближенные к реальности. Таким образом реализуется основная цель, продемонстрировать уровень готовности технологии на модельной среде максимально приближенной к реальности, а также проверить соответствие технологии требованиям к производительности (провести профилирование на реальных объемах и убедиться в эффективности процедуры масштабирования).

Кейс 1. Генеративный ИИ для автоматического составления инвестиционных обзоров

Описание: Аналитики банка ежедневно готовят десятки аналитических и инвестиционных обзоров по рынкам, компаниям, макроэкономическим показателям. Задача – исследовать применение больших языковых моделей (LLM) для генерации кратких сводок и аналитических отчетов на основе входных данных (биржевые котировки, макроэкономические показатели, рыночные события). По сути, требуется автоматизировать подготовку инвестиционного обзора, чтобы экономить время аналитиков и обеспечить единообразие стиля.

Цель: Разработать инструмент на базе LLM, способный по структурированным данным и краткому описанию формировать инвестиционный обзор в деловом банковском стиле. Иными словами, настроить промпт и модель так, чтобы на основе табличных данных и краткой справки генерировался связный текст отчёта.

Ожидаемый результат: Прототип модели, генерирующей аналитический текст обзора объемом 500–1000 слов с разделами «Обзор событий», «Рекомендации», «Прогнозы», оформленный по стандартам банка. Студент должен представить пример сгенерированного обзора и показать, как промпт-инжиниринг позволил получить осмысленный структурированный отчет.

Кейс 2. NLP-анализ жалоб клиентов в свободной форме

Описание: В клиентскую службу банка поступают обращения от клиентов через чат, мобильное приложение и форму жалоб. Эти сообщения пишутся в свободной форме и могут содержать описания проблем, эмоции клиентов и т.д. Необходимо построить NLP-модель семантического анализа, которая с помощью методов промпт-инжиниринга выделяет суть обращения (основную проблему клиента), определяет тональность (эмоциональную окраску) и оценивает потенциальную серьезность инцидента. Это поможет банку быстрее реагировать на проблемы и выявлять типовые болевые точки.

Цель: Автоматизировать первичную классификацию клиентских обращений для ускорения их маршрутизации и выявления повторяющихся проблем в продуктах или процессах. С помощью LLM планируется распознавать ключевые темы жалобы (например, сбой приложения, ошибка платежа) и эмоцию клиента, чтобы решать вопрос приоритетности ответа.

Ожидаемый результат: Прототип решения (например, на базе LLM или специализированной модели), который автоматически извлекает из текста обращения тему жалобы (например, «двойное списание», «не работает карта»), определяет эмоциональную тональность (негативная, нейтральная, позитивная) и степень критичности. Студент предоставит примеры анализа нескольких жалоб и покажет, как промпты настроены для извлечения этих сведений.

Кейс 3. Мультимодальный ассистент для банковского отделения

Описание: В физических отделениях банка планируется внедрение интерактивных ИИ-консультантов для помощи клиентам. Предполагается создать прототип мультимодального ассистента, который умеет воспринимать речь и визуально ориентироваться в отделении. Например, такой ассистент должен распознавать лица клиентов или предъявленные ими документы, видеть расположение банкоматов и при этом понимать голосовые запросы. Он комбинирует компьютерное зрение и обработку естественного языка, отвечая голосом на вопросы клиентов и давая визуальные подсказки.

Цель: Разработать прототип системы, имитирующей функциональность такого помощника. Это включает: ответ на типовые вопросы клиентов голосом, распознавание и описание визуальных объектов (например, отсканировать QR-код паспорта или показать путь до нужного окна), навигацию клиента внутри отделения и т.д.. Промпт-инжиниринг здесь нужен для координации разных модальностей (текстовой и визуальной) и формата ответов ассистента.

Ожидаемый результат: Интерактивная мультимодальная модель (или эмуляция её работы), которая на запрос пользователя голосом может выполнить действие: показать нужный участок отделения на экране, прокомментировать прогресс обслуживания, указать на ближайший банкомат и т.п.. Студент демонстрирует работу ассистента на нескольких сценариях (например, клиент спрашивает голосом, ассистент реагирует словами и отображает информацию на экране).

Кейс 4. Оптимизация промптов и оценка их качества в банковском чат-боте

Описание: Банк внедряет AI-чат-бот для консультирования клиентов по продуктам и услугам. Качество ответов такого бота сильно зависит от того, как сформулированы системные и пользовательские промпты (инструкции для LLM). В этом кейсе студентам предлагается проанализировать, как различные формулировки промпта влияют на ответы модели, и выработать методику оценки качества промптов. Например, короткий неопределённый запрос может приводить к расплывчатому ответу, тогда как структурированный промпт с указанием ролей и формата ответа – к более точному и полезному результату. Необходимо уметь измерять качество ответов модели (наличие ошибок, полнота ответа, соответствие инструкции и ожиданиям) и на основе этого улучшать промпты.

Цель: Разработать и протестировать подход к улучшению промптов для банковского чат-бота. Студент должен подобрать метрики качества ответов (например, доля правильных решений задач пользователя, удовлетворённость пользователей ответами, отсутствие «галлюцинаций» в ответе) и на их основе сравнить несколько вариантов промптов. Цель – добиться того, чтобы доля верных и полезных ответов выросла до целевого уровня за счёт оптимизации формулировок запросов и инструкций для LLM.

Ожидаемый результат: Отчёт или мини-исследование, демонстрирующее оценку качества разных вариантов промптов и финальную оптимизированную версию. Например,

студент может привести таблицу с метриками (релевантность ответа, фактическая корректность, соблюдение стиля и т.д.) для промпта версии А vs версии В, и показать, что после улучшения промпта качество ответов повысилось. Ожидается, что студент предложит рекомендации по формированию эффективных промптов (структурирование инструкции, указание контекста, примеров и пр.) на основе проведённых экспериментов.

Кейс 5: (CarX TECHNOLOGIES). Прототипирование MVP гоночной аркады с использованием генеративного ИИ

Контекст. Студия CarX TECHNOLOGIES хочет быстро проверить новую концепцию уличной гоночной аркады с элементами тюнинга. Задача: за 2 недели создать минимально жизнеспособный продукт (MVP), который можно показать фокус-группе. В команде нет художников и звукорежиссёров – только разработчики, владеющие промпт-инжинирингом и интеграцией ИИ. Разрешено использовать любые генеративные модели для создания ассетов и фрагментов кода.

Задание (кейс).

- Разработайте промпты для генерации: (а) трёх фоновых треков (дрифт, стрит-рейсинг, ночной город) через LLM (описание для аудио-модели); (б) набора текстур дорожного покрытия и авто через YandexART; (в) базовых скриптов управления автомобилем и ИИ противника на промпте «создай C# скрипт для следования по трассе с поворотами».
- Опишите архитектуру сборки MVP: какие компоненты генерируются ИИ, какие пишутся вручную.
- Предложите критерии оценки качества сгенерированного контента (например, соответствие стилю, частота ошибок в коде).
- Напишите системный промпт для агента, который помогает отбирать лучшие варианты текстур из 10 сгенерированных.
- Оцените временные затраты: сколько человеко-часов сэкономило использование ИИ по сравнению с традиционной разработкой.

Ожидаемый результат:

- дорожная карта создания MVP с указанием used AI tools;
- примеры промптов и сгенерированных результатов (скриншоты, фрагменты кода);
- оценка эффективности (ROI) применения ИИ в прототипировании.

Кейс 6: (CarX TECHNOLOGIES). Обучение RL-агента для задачи «догони лидера» в физической симуляции

Контекст. В игре CarX Street игроки жалуются на слишком предсказуемых противников. Нужен адаптивный AI-соперник, который учится преследовать лидера гонки, используя реальную физику автомобиля (инерция, сцепление с дорогой). Команда уже имеет симуляционную среду на Unity ML-Agents. Требуется обучить агента, который может догонять и удерживаться за целевым автомобилем на трассе с поворотами, не вылетая.

Роль студента. ML-инженер, настраивающий среду, функцию награды и параметры обучения PPO.

Задание (кейс).

- Опишите пространство наблюдений (observations) для агента: какие параметры (относительная скорость, угол, расстояние, текущая скорость, показания датчиков касания дороги) необходимо передавать.
- Предложите функцию награды (reward function) на каждом шаге, которая поощряет: сокращение дистанции до лидера, нахождение в зоне «след в след» (tailgating), прохождение контрольных точек; и штрафует: вылет с трассы, столкновение, слишком большое отставание.
- Какие действия (discrete/continuous) должны быть у агента: газ/тормоз/руль? Обоснуйте.

- Опишите конвейер обучения: как собирать опыт, как часто обновлять политику, как избегать переобучения под конкретную трассу.
- Предложите метод оценки качества обученного агента (например, среднее время удержания лидера, количество обгонов за 5 кругов).
- Напишите псевдокод цикла сбора данных для обучения с подкреплением (agent step → environment step → reward → buffer).

Ожидаемый результат:

- спецификация среды и RL-параметров;
- графики обучения (expected reward per episode);
- видеодемонстрация поведения обученного агента (описание того, что должно быть видно);
- рекомендации по развёртыванию модели в рантайме игры.

7. Материально-техническое обеспечение по дисциплине (модулю)

Виртуальные машины, кластер Managed Kubernetes и ресурсы GPU в облаке предоставляется индустриальным партнером ПАО «Сбербанк»:

№	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

Дополнительные облачные ресурсы предоставляются технологическим партнером Yandex Cloud.

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, обеспечивающее доступ к сети интернет (проводное соединение и беспроводное соединение по технологии Wi-Fi)	MS Office Word 2016 и выше Ms Power Point 2016 и выше

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)
--	--	--

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная учебная работа (консультации) – дополнительное разъяснение учебного материала.

Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или лицом с ограниченными возможностями здоровья.

Примечание: Конкретизация аудиторий и их оснащение определяется ОПОП.