

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

УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования – первый
проректор

Хагуров Т.А.

« 29 » августа 2025 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1. В.19 Генеративный искусственный интеллект

Направление подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем

Профиль Искусственный интеллект и аналитика данных

Форма обучения очная

Квалификация бакалавр

Краснодар 2025

Рабочая программа дисциплины «Генеративный искусственный интеллект» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем.

Программу составил(и):

Е.В. Казаковцева, доцент, канд. физ.-мат. наук



Рабочая программа дисциплины утверждена на заседании центра искусственного интеллекта

протокол № 01 «28» августа 2025 г.

Руководитель центра ИИ Коваленко А.В.



подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол № 01 «28» августа 2025 г.

Председатель УМК факультета Коваленко А.В.



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 глубокого понимания принципов функционирования моделей генеративного искусственного интеллекта, изучение методов и технологий генерации мультимодальных данных, освоение современных подходов проектирования и тренировки генеративных моделей, необходимых для эффективной профессиональной деятельности специалистов по обработке естественного языка, компьютерному зрению, дизайну и созданию инновационных решений на основе генеративных нейросетей.

1.2 Задачи дисциплины

1. Изучение теоретических основ генеративных моделей искусственного интеллекта: архитектуры генераторов, дискриминаторов, автокодировщиков, трансформеров, диффузионных моделей и др.

2. Освоение ключевых концепций и методологий обучения генеративных моделей: вариационные автоэнкодеры (VAE), генеративно-состязательные сети (GANs), обучение с учителем и без учителя, перцептивное качество генерируемых объектов.

3. Развитие практических навыков построения и тренировки генеративных моделей на примере популярных библиотек и фреймворков (PyTorch, Hugging Face Transformers, Optuna, Hugging Face Diffusers и т.д.).

4. Владение методами оценки качества и точности генерируемого контента, такими как метрики FID, PSNR, SSIM, BLEU и др.

5. Формирование компетенций по применению генеративных моделей в прикладных областях: создание реалистичных изображений, синтез речи, автоматическая генерация текста, разработка виртуальных персонажей и интерфейсов взаимодействия.

6. Обзор этических аспектов и проблем безопасности, связанных с использованием генеративных моделей ИИ (deepfakes, манипуляции информацией, конфиденциальность данных).

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Генеративный искусственный интеллект» относится к части, формируемой участниками образовательных отношений Блока 1 «Дисциплины (модули)» учебного плана. Для успешного освоения данной дисциплины необходимы знания следующих дисциплин: Математический анализ, Векторная алгебра, Основы программирования, Обработка данных на Python, Математические модели нейронных сетей, Нейросетевые технологии, Технологии обработки языка, звуковых данных, включая распознавание и синтез речи.

1.4 Профессиональные роли в структуре образовательной программы

Роль 1: Data Analyst (Аналитик данных)

Задачи:

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

Роль 2: MLOps (Специалист по эксплуатации ИИ)

Задачи:

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

Роль 3: AI PM (Менеджер проектов ИИ)

Задачи:

- 1. Управление ИИ-проектами от идеи до внедрения*

2. Анализ бизнес-требований и постановка задач

3. Оценка эффективности и ROI ИИ-решений

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

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

| Код, уровень и формулировка компетенции | Индикаторы | Уровни освоения индикаторов компетенции |
|---|--|--|
| LLM-1 Способен применять и (или) разрабатывать генеративные модели и БЯМ | LLM-1.1 Знает архитектуры генеративных моделей | (Б) Отличает виды генеративных моделей и принципы их действия |
| | LLM-1.7 Проводит валидацию и тестирование генеративных моделей | (Б) Применяет ручную оценку качества |
| FC-2 Способен проводить фронтальные исследования в области фундаментальных и генеративных моделей | FC-2.1 Исследует и разрабатывает большие языковые модели (LLM) и другие модели для символьных данных | (Б) Владеет принципами работы систем на базе символьного искусственного интеллекта. Умеет использовать готовые нейросимволических фреймворков DeepProbLog, Neurosymbolic AI Toolkit |
| DL-1 Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей | DL-1.1 Способен объяснять и применять математические основы нейронных сетей, включая расчет градиентов, методы оптимизации и алгоритм обратного распространения ошибки (backpropagation), для эффективного обучения моделей. | (П) задает скорость обучения в зависимости от задачи и набора данных; выбирает функцию потерь в зависимости от задачи и набора данных; способен применять регуляризацию и прореживание; выбирает размер пакета для стохастического градиентного спуска; понимает принцип градиентного спуска |
| | DL-1.3 Способен применять современные архитектуры глубоких сетей для решения различных задач, понимая их внутреннюю структуру и особенности обучения. | (П) Применяет принцип построения вычислительного блока Google Inception; Разрабатывает решения с применением backbone сетей; Знает отличия и способен применять нейронные сети для отслеживания объектов (семейство R-CNN, YOLO) |
| DL-2 Способен применять и (или) разрабатывать современные архитектуры | DL-2.1 Применяет известные архитектуры генеративных глубоких нейронных сетей для | (Б) Умеет использовать популярные генеративные модели (GPT, Stable Diffusion, VQ-VAE) через |

| | | |
|--|---|---|
| генеративных глубоких сетей | решения прикладной задачи (генерация текста, генерация изображений по тексту, синтез речи и т.д.), при необходимости проводя дообучение на наборах данных | API или готовые реализации. Запускает инференс на стандартных задачах (генерация текста по промпту, создание изображений). Работает с базовыми параметрами генерации (temperature, top-k sampling). Подготавливает данные для дообучения (токенизация текста, нормализация изображений). Форматирует данные под требования модели (например, промпты для тексто-изображение моделей). |
| DL-4 Способен применять и (или) разрабатывать алгоритмы, методы и технологии обработки естественного языка | DL-4.1 Применяет (проводя выбор и эксперименты) известные алгоритмы и библиотеки для обработки естественного языка, предобученные глубокие нейросетевые модели для прикладных задач анализа текстов, при необходимости дообучая и валидируя на собственных наборах данных | (П) Владеет инструментами грамматического разбора структурированных и слабо-структурированных текстов, способен написать свой парсер. Владеет инструментами разметки текстовых данных и формирования словарей. |

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 3 зач. ед. (108 часов), их распределение по видам работ представлено в таблице

| Вид учебной работы | Всего часов | Семестры (часы) | | | | | |
|--|-------------|-----------------|--|--|--|--|--|
| | | 8 | | | | | |
| Контактная работа, в том числе: | 44.3 | 44.3 | | | | | |
| Аудиторные занятия (всего): | 42 | 42 | | | | | |
| Занятия лекционного типа | 14 | 14 | | | | | |
| Лабораторные занятия | 28 | 28 | | | | | |
| Занятия семинарского типа (семинары, практические занятия) | | | | | | | |
| Иная контактная работа: | 2.3 | 2.3 | | | | | |
| Контроль самостоятельной работы (КСР) | 2 | 2 | | | | | |
| Промежуточная аттестация (ИКР) | 0.3 | 0.3 | | | | | |
| Самостоятельная работа, в том числе: | 10 | 10 | | | | | |
| Курсовая работа | - | - | | | | | |
| Проработка учебного (теоретического) материала | 6 | 6 | | | | | |

| | | | | | | | |
|---|--------------------------------------|-------------|-------------|--|--|--|--|
| Выполнение индивидуальных заданий (подготовка сообщений, презентаций) | - | - | | | | | |
| Реферат | - | - | | | | | |
| Подготовка к текущему контролю | 4 | 4 | | | | | |
| Контроль: | 53,7 | 53,7 | | | | | |
| Подготовка к экзамену | 53,7 | 53,7 | | | | | |
| Общая трудоемкость | час. | 108 | 108 | | | | |
| | в том числе контактная работа | 44.3 | 44.3 | | | | |
| | зач. Ед | 3 | 3 | | | | |

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.
Разделы (темы) дисциплины, изучаемые в 8 семестре

| № | Наименование разделов (тем) | Всего | Количество часов | | | |
|---|---|------------|-------------------|----|-----------|----------------------|
| | | | Аудиторная работа | | | Внеаудиторная работа |
| | | | Л | ПЗ | ЛР | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1. | Основные концепции и типы генеративных моделей | 5 | 2 | | 2 | 1 |
| 2. | Генеративные нейросети | 14 | 2 | | 10 | 2 |
| 3. | Мультимодальность и трансформеры | 12 | 4 | | 6 | 2 |
| 4. | Технологии и приложения мультимодального генеративного ИИ | 10 | 2 | | 6 | 2 |
| 5. | Методы оценки качества генеративных моделей | 5 | 2 | | 2 | 1 |
| 6. | Современные исследования и практические кейсы | 6 | 2 | | 2 | 2 |
| ИТОГО по разделам дисциплины | | 52 | 14 | | 28 | 10 |
| Контроль самостоятельной работы (КСР) | | 2 | | | | |
| Промежуточная аттестация (ИКР) | | 0.3 | | | | |
| Подготовка к экзамену | | 53,7 | | | | |
| Общая трудоемкость по дисциплине | | 108 | | | | |

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

2.3 Содержание разделов (тем) дисциплины

2.3.1 Занятия лекционного типа

| № | Наименование раздела (темы) | Содержание раздела (темы) | Форма текущего контроля |
|----|--|---|-------------------------|
| 1 | 2 | 3 | 4 |
| 1. | Основные концепции и типы генеративных моделей | Тема: Определение и классификация генеративных моделей. | Т, ЛР |

| № | Наименование раздела (темы) | Содержание раздела (темы) | Форма текущего контроля |
|----|----------------------------------|--|-------------------------|
| 1 | 2 | 3 | 4 |
| | | <p><i>Тема: Общие подходы к обучению генеративных моделей.</i></p> <p><i>Тема: Введение в семантику данных и представление признаков. Программное обеспечение для НС.</i></p> | |
| 2. | Генеративные нейросети | <p><i>Тема: Автокодировщики (Autoencoders):</i></p> <ul style="list-style-type: none"> ○ Базовая структура и алгоритм работы автокодировщика. ○ Задача реконструкции и латентное пространство. <p><i>Тема: Вариационные автокодировщики (VAEs):</i></p> <ul style="list-style-type: none"> ○ Вероятностный подход к задаче кодирования. ○ Латентное распределение и механизм декодирования. <p><i>Тема: Диффузионные модели (Diffusion Models):</i></p> <ul style="list-style-type: none"> ○ Шумы и обратный процесс формирования изображения. ○ Примеры реализации (Stable Diffusion, Latent Diffusion Model). <p><i>Тема: Генеративно-сопоставительные сети (GANs):</i></p> <ul style="list-style-type: none"> ○ Архитектура и принцип работы GAN. ○ Известные варианты (DCGAN, StyleGAN, Pix2Pix). <p><i>Тема: Архитектуры типа Expert-in-the-loop (MoE):</i></p> <ul style="list-style-type: none"> ○ Гибридные системы и концепция экспертов. ○ Использование гибридных архитектур для повышения эффективности. | Т, ЛР |
| 3. | Мульти-modalность и трансформеры | <p><i>Тема: Концепция мульти-modalности: интеграция визуальной, текстовой и звуковой информации.</i></p> <p><i>Тема: Multimodal Transformers:</i></p> <ul style="list-style-type: none"> ○ Общая архитектура мультиспектральных трансформеров. ○ Особенности внимания (attention mechanisms) и смешивание данных разных модальностей. <p><i>Тема: Практики применения мульти-modalных трансформеров:</i></p> <ul style="list-style-type: none"> ○ Image Captioning (описание изображений текстом). | Т, ЛР |

| № | Наименование раздела (темы) | Содержание раздела (темы) | Форма текущего контроля |
|----|---|--|-------------------------|
| 1 | 2 | 3 | 4 |
| | | <ul style="list-style-type: none"> ○ Speech-to-text и text-to-speech конвертеры. <p><i>Тема: Применение в приложениях Cross-modal learning:</i></p> <ul style="list-style-type: none"> ○ Клиппинг (CLIP) и русскоязычные аналоги (ruCLIP). ○ Интерактивные нейроархитектуры (Flamingo). | |
| 4. | Технологии и приложения мультимодального генеративного ИИ | <p><i>Тема: Использование готовых моделей из публичных репозиториев:</i></p> <ul style="list-style-type: none"> ○ Платформы хранения моделей (Hugging Face, GitHub). ○ Предобученные модели для решения конкретных задач (Speech-to-text, Text-to-image). <p><i>Тема: Изменение гиперпараметров моделей под конкретные задачи:</i></p> <ul style="list-style-type: none"> ○ Выбор значений sample rate, batch size и other hyperparameters. ○ Методика подбора оптимальных настроек. <p><i>Тема: Предобученные модели для распознавания и генерации речи:</i></p> <ul style="list-style-type: none"> ○ Whisper, Wav2Vec, Tacotron. ○ Достоинства и ограничения каждого инструмента. | Т, ЛР |
| 5. | Методы оценки качества генеративных моделей | <p><i>Тема: Метрика Maximum Mean Discrepancy (MMD):</i></p> <ul style="list-style-type: none"> ○ Принцип расчета расстояния между распределениями. ○ Интерпретация результата. <p><i>Тема: Fréchet Inception Distance (FID):</i></p> <ul style="list-style-type: none"> ○ Расчет близости распределения реального и синтетического изображений. ○ Значимость показателя в оценке качества изображений. <p><i>Тема: Inception Score (IS):</i></p> <ul style="list-style-type: none"> ○ Как измеряется разнообразие и четкость изображений. ○ Связь IS с качеством восприятия человеком. <p><i>Тема: Perplexity:</i></p> <ul style="list-style-type: none"> ○ Измерение сложности предсказания последовательности символов. | Т, ЛР |

| № | Наименование раздела (темы) | Содержание раздела (темы) | Форма текущего контроля |
|----|---|--|-------------------------|
| 1 | 2 | 3 | 4 |
| | | <ul style="list-style-type: none"> ○ Применение в NLP-моделях и оценка quality generation. <p><i>Тема: Оценка разнообразия (Diversity):</i></p> <ul style="list-style-type: none"> ○ Вычисления показателей диверсификации выборки. ○ Важность измерения diversity для предотвращения overfitting. | |
| 6. | Современные исследования и практические кейсы | <p><i>Тема: Реализация сложных генеративных задач:</i></p> <ul style="list-style-type: none"> ○ Комплексные проекты с использованием нескольких модулей (audio + visual + text). ○ Реализации сценариев совместной генерации аудио и видеопотоков. <p><i>Тема: RAG (Retrieval-Augmented Generation):</i></p> <ul style="list-style-type: none"> ○ Совместное использование извлечения информации и генеративных механизмов. ○ Примеры практического применения RAG в коммерческих продуктах. <p><i>Тема: Case-study практики компаний и стартапов:</i></p> <ul style="list-style-type: none"> ○ Опыт интеграции генеративных моделей в коммерческие продукты. ○ Текущие тенденции развития индустрии генеративного ИИ. | Т, ЛР |

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

2.3.2 Занятия семинарского типа

Не предусмотрены

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.

2.3.3 Лабораторные занятия

| № | Наименование раздела (темы) | Наименование лабораторных работ | Форма текущего контроля |
|----|---|---|-------------------------|
| 1 | 2 | 3 | 4 |
| 1. | Основные концепции и типы генеративных моделей | Лабораторная работа №1: Введение в генеративные модели | ЛР |
| 2. | Генеративные нейросети | Лабораторная работа №2: Работа с автокодировщиками (Autoencoders) Лабораторная работа №3: Реализация вариационного автокодировщика (VAE) Лабораторная работа №4: Диффузионные модели (Diffusion Models) Лабораторная работа №5: Генеративно-состязательные сети (GANs) | ЛР |
| 3. | Мультимодальность и трансформеры | Лабораторная работа №6: Introduction to Multimodal Data Processing Лабораторная работа №7: Реализация Multimodal Transformers | ЛР |
| 4. | Технологии и приложения мультимодального генеративного ИИ | Лабораторная работа №8: Генераторы текста и речи Лабораторная работа №9: Гиперпараметризация моделей | ЛР |
| 5. | Методы оценки качества генеративных моделей | Лабораторная работа №10: Анализ метрик генеративных моделей | ЛР |
| 6. | Современные исследования и практические кейсы | Лабораторная работа №11: Case Study in Retrieval-Augmented Generation (RAG) | ЛР |

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

2.3.4 Примерная тематика курсовых работ (проектов)

Не предусмотрены

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

| № | Вид СРС | Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы |
|---|-----------------------------------|--|
| 1 | 2 | 3 |
| 1 | Изучение теоретического материала | Методические указания по организации самостоятельной работы студентов, утвержденные УСФ, протокол №1 от 30.06.2025 |
| 2 | Решение задач | Методические указания по организации самостоятельной работы студентов, утвержденные УСФ, протокол №1 от 30.06.2025 |

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.

| Семестр | Вид занятия | Используемые интерактивные образовательные технологии | количество интерактивных часов |
|--------------|-------------|---|--------------------------------|
| 8 | ЛР | Практические занятия в режимах взаимодействия «преподаватель – студент» и «студент – студент» | 8 |
| Итого | | | 8 |

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

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 «Генеративный искусственный интеллект».

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

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

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

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

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

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

Для лиц с нарушениями зрения:

– в печатной форме увеличенным шрифтом,

– в форме электронного документа.

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

Структура оценочных средств для текущей и промежуточной аттестации

| № п/п | Контролируемые разделы (темы) дисциплины | Код контролируемой компетенции (или ее части) | Наименование оценочного средства | |
|-------|--|---|--|--------------------------|
| | | | Текущий контроль | Промежуточная аттестация |
| 1 | Основные концепции и типы генеративных моделей | LLM-1.1; LLM-1.7 | Тестирование, Лабораторная работа №1 | Вопросы к экзамену 1-7 |
| 2 | Генеративные нейросети | LLM-1.1; LLM-1.7; FC-2.1; DL- | Тестирование, Лабораторные работы №2-5 | Вопросы к экзамену 8-14 |

| | | | | |
|---|---|--|---|--------------------------|
| | | 1.1; DL-1.3; DL-2.1; DL-4.1 | | |
| 3 | Мультимодальность и трансформеры | FC-2.1; DL-1.1; DL-1.3; DL-2.1; DL-4.1 | Тестирование, Лабораторные работы №6, 7 | Вопросы к экзамену 15-21 |
| 4 | Технологии и приложения мультимодального генеративного ИИ | LLM-1.1; LLM-1.7; FC-2.1; DL-1.1; DL-1.3; DL-2.1; DL-4.1 | Тестирование, Лабораторные работы №8, 9 | Вопросы к экзамену 22-27 |
| 5 | Методы оценки качества генеративных моделей | LLM-1.1; LLM-1.7 | Тестирование, Лабораторная работа №10 | Вопросы к экзамену 28-33 |
| 6 | Современные исследования и практические кейсы | FC-2.1 | Тестирование, Лабораторная работа №11 | Вопросы к экзамену 34-39 |

Показатели, критерии и шкала оценки сформированных компетенций

Соответствие **пороговому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **удовлетворительно**):

LLM-1 Б *Способен применять и (или) разрабатывать генеративные модели и БЯМ*

Отличает виды генеративных моделей и принципы их действия
Применяет ручную оценку качества

FC-2 Б *Способен проводить фронтальные исследования в области фундаментальных и генеративных моделей*

Владеет принципами работы систем на базе символьного искусственного интеллекта. Умеет использовать готовые нейро-символических фреймворков DeepProbLog, Neurosymbolic AI Toolkit

DL-1 П *Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей*

Выбирает функцию активации и степень связности нейронов в зависимости от задачи; применяет обратное распространение ошибки для обновления весов нейронов; выбирает способ формирования начальных значений весов нейронов
Выбирает функцию активации и степень связности нейронов в зависимости от задачи; применяет обратное распространение ошибки для обновления весов нейронов; выбирает способ формирования начальных значений весов нейронов

DL-2 Б *Способен применять и (или) разрабатывать современные архитектуры генеративных глубоких сетей*

Умеет использовать популярные генеративные модели (GPT, Stable Diffusion, VQ-VAE) через API или готовые реализации. Запускает инференс на стандартных задачах (генерация текста по промпту, создание изображений). Работает с базовыми параметрами генерации (temperature, top-k sampling). Подготавливает данные для дообучения (токенизация текста, нормализация изображений). Форматирует данные под требования модели (например, промпты для тексто-изображение моделей).

- DL-4 П** *Способен применять и (или) разрабатывать алгоритмы, методы и технологии обработки естественного языка*
Владеет классическими инструментами парсинга текстов: регулярные выражения, токенизация, морфологический анализ, синтаксический анализ

Соответствие **базовому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **хорошо**):

- LLM-1 Б** *Способен применять и (или) разрабатывать генеративные модели и БЯМ*

Сравнивает архитектуры и выбирает подходящую под задачу
Автоматизирует валидацию на основе тестов и метрик

- FC-2 Б** *Способен проводить фронтальные исследования в области фундаментальных и генеративных моделей*

Умеет разрабатывать custom symbolic layers для PyTorch. Умеет интегрировать различные формальные верификаторы Z3, Prolog в пайплайны генерации

- DL-1 П** *Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей*

задает скорость обучения в зависимости от задачи и набора данных; выбирает функцию потерь в зависимости от задачи и набора данных; способен применять регуляризацию и прореживание; выбирает размер пакета для стохастического градиентного спуска; понимает принцип градиентного спуска

Применяет принцип построения вычислительного блока Google Inception; Применяет принцип работы блока остатка в ResNet; Разрабатывает решения с применением backbone сетей; Знает отличия и способен применять нейронные сети для отслеживания объектов (семейство R-CNN, YOLO)

- DL-2 Б** *Способен применять и (или) разрабатывать современные архитектуры генеративных глубоких сетей*

Настраивает параметры генерации под конкретную задачу. Реализует техники контролируемой генерации (guidance scale, prompt engineering). Комбинирует несколько моделей в пайплайн. Проводит domain adaptation для специфических данных. Использует продвинутые методы эффективного обучения (QLoRA, DreamBooth). Оптимизирует процесс обучения (подбор lr, батчей).

- DL-4 П** *Способен применять и (или) разрабатывать алгоритмы, методы и технологии обработки естественного языка*

Владеет инструментами грамматического разбора структурированных и слабо-структурированных текстов, способен написать свой парсер. Владеет инструментами разметки текстовых данных и формирования словарей.

Соответствие **продвинутому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **отлично**):

- LLM-1 Б** *Способен применять и (или) разрабатывать генеративные модели и БЯМ*

Отличает виды генеративных моделей и принципы их действия
Сравнивает архитектуры и выбирает подходящую под задачу
Применяет ручную оценку качества

Автоматизирует валидацию на основе тестов и метрик

- FC-2 Б** *Способен проводить фронтальные исследования в области фундаментальных и генеративных моделей*

Владеет принципами работы систем на базе символьного искусственного интеллекта. Умеет использовать готовые нейро-символических фреймворков DeepProbLog, Neurosymbolic AI Toolkit

Умеет разрабатывать custom symbolic layers для PyTorch. Умеет интегрировать различные формальные верификаторы Z3, Prolog в пайплайны генерации

DL-1 П *Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей*

Выбирает функцию активации и степень связности нейронов в зависимости от задачи; применяет обратное распространение ошибки для обновления весов нейронов; выбирает способ формирования начальных значений весов нейронов

Применяет принцип работы блока остатка в ResNet; Регулирует поток вычисления градиента в глубоких нейронных сетях

задает скорость обучения в зависимости от задачи и набора данных; выбирает функцию потерь в зависимости от задачи и набора данных; способен применять регуляризацию и прореживание; выбирает размер пакета для стохастического градиентного спуска; понимает принцип градиентного спуска

Выбирает функцию активации и степень связности нейронов в зависимости от задачи; применяет обратное распространение ошибки для обновления весов нейронов; выбирает способ формирования начальных значений весов нейронов

Применяет принцип построения вычислительного блока Google Inception;

Применяет принцип работы блока остатка в ResNet; Разрабатывает решения с применением backbone сетей; Знает отличия и способен применять нейронные сети для отслеживания объектов (семейство R-CNN, YOLO)

DL-2 Б *Способен применять и (или) разрабатывать современные архитектуры генеративных глубоких сетей*

Умеет использовать популярные генеративные модели (GPT, Stable Diffusion, VQ-VAE) через API или готовые реализации. Запускает инференс на стандартных задачах (генерация текста по промпту, создание изображений).

Работает с базовыми параметрами генерации (temperature, top-k sampling).

Подготавливает данные для дообучения (токенизация текста, нормализация изображений). Форматирует данные под требования модели (например, промпты для тексто-изображение моделей).

Настраивает параметры генерации под конкретную задачу. Реализует техники контролируемой генерации (guidance scale, prompt engineering).

Комбинирует несколько моделей в пайплайн. Проводит domain adaptation для специфических данных. Использует продвинутые методы эффективного обучения (QLoRA, DreamBooth). Оптимизирует процесс обучения (подбор lr, батчей).

DL-4 П *Способен применять и (или) разрабатывать алгоритмы, методы и технологии обработки естественного языка*

Владеет классическими инструментами парсинга текстов: регулярные выражения, токенизация, морфологический анализ, синтаксический анализ

Владеет инструментами грамматического разбора структурированных и слабо-структурированных текстов, способен написать свой парсер. Владеет инструментами разметки текстовых данных и формирования словарей.

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 тестирования

Вопрос 1: Что такое генеративная модель?

- A) Модель, предназначенная исключительно для классификации.
- B) Нейросеть, специализирующаяся на сегментации изображений.
- C) Модель, создающая новые данные, аналогичные имеющимся.
- D) Система обнаружения аномалий.

Вопрос 2: Какая из перечисленных моделей не относится к классу генеративных моделей?

- A) Restricted Boltzmann Machines (RBMs)
- B) Generative Adversarial Networks (GANs)
- C) Logistic Regression
- D) Variational Autoencoders (VAEs)

Вопрос 3: Что такое латентное пространство в контексте генеративных моделей?

- A) Прямой выход модели, содержащий истинные метки.
- B) Низкоресурсное пространство, в котором хранится обработанная информация.
- C) Высокая размерность входных данных.
- D) Скрытая область, отражающая внутреннюю структуру данных.

Вопрос 4: Какая функция лежит в основе работы вариационного автокодировщика (VAE)?

- A) Минимизация ошибки градиента.
- B) Сохранение максимальной энтропии.
- C) Максимальная вероятность априорного распределения скрытых признаков.
- D) Ограничение количества слоёв в сети.

Вопрос 5: В каком порядке строится процесс генерации изображения в диффузионных моделях?

- A) Постепенное удаление шума из случайно зашумлённых данных.
- B) Непосредственный перенос информации из нормального распределения.
- C) Случайная инициализация пикселей и последующая минимизация потерь.
- D) Только прямое копирование существующих изображений.

Вопрос 6: Что значит архитектура GAN (Generative Adversarial Network)?

- A) Генеративная сеть, создающая только двоичную классификацию.
- B) Два конкурирующих компонента: генератор и дискриминатор.
- C) Простая сверточная сеть без дополнительного обучения.
- D) Группа независимых моделей, работающих отдельно друг от друга.

Вопрос 7: Какую важную проблему вызывает стандартная процедура обучения GAN?

- A) Легкость обучения и высокая устойчивость.
- B) Проблема нестабильного обучения и колебание потерь.
- C) Отсутствие необходимости в большом объеме данных.
- D) Быстрое достижение глобального минимума.

Вопрос 8: Что такое мультимодальность в контексте генеративных моделей?

- A) Передача данных от одной модели к другой.
- B) Одновременная работа с данными различных форматов (текст, изображение, звук).
- C) Повторное использование одних и тех же данных для разных задач.
- D) Упрощённая форма сверточных нейросетей.

Вопрос 9: Что такое Image Captioning?

- A) Модель, умеющая синтезировать человеческую речь.
- B) Метод обработки речи в текст.
- C) Задача генерации текста, описывающего изображение.
- D) Автоматическое разделение речи на отдельные звуки.

Вопрос 10: Что такое модель CLIP (Contrastive Language–Image Pre-training)?

- A) Нейросеть для определения местоположения объектов на карте.
- B) Архитектура, совмещающая визуальные и текстовые признаки.
- C) Система для выявления токсичного поведения в социальных сетях.
- D) Платформа для анализа рынка ценных бумаг.

Вопрос 11: Что такое предобученные модели?

- A) Модели, специально адаптированные для конкретной задачи.
- B) Сети, изначально создаваемые для узких нишевых задач.
- C) Модели, заранее натренированные на крупных общедоступных датасетах.
- D) Машины, работающие автономно без участия оператора.

Вопрос 12: Зачем нужно изменять гиперпараметры (hyperparameters) в генеративных моделях?

- A) Чтобы ускорить обучение и снизить стоимость вычислений.
- B) Чтобы избежать переобучения и повысить обобщаемость модели.
- C) Потому что изменение гиперпараметров никак не влияет на результат.
- D) Для сокращения объема памяти, необходимой для обучения.

Вопрос 13: Какая модель чаще всего используется для преобразования текста в речь?

- A) CNN (Convolutional Neural Network)
- B) WaveNet
- C) PCA (Principal Component Analysis)
- D) SVM (Support Vector Machine)

Вопрос 14: Какая проблема возникает при плохом подборе гиперпараметров в генеративных моделях?

- A) Чрезмерно быстрая генерация данных.
- B) Недообучение или переобучение модели.
- C) Снижение стоимости обучения.
- D) Увеличение объемов памяти.

Вопрос 15: Что измеряет метрика FID (Fréchet Inception Distance)?

- A) Расстояние между средним изображением и набором реальных данных.
- B) Уровень сходства распределения сгенерированных и реальных данных.
- C) Размер стандартного отклонения сгенерированного набора данных.
- D) Средний процент ошибки классификации.

Вопрос 16: Что означает низкая величина метрики Perplexity в задачах NLP?

- A) Высокая сложность генерации текста.
- B) Медленная скорость генерации текста.
- C) Хорошее качество и согласованность сгенерированного текста.
- D) Высокий уровень шума в генерации.

Вопрос 17: Что оценивает метрика Diversity?

- A) Общую длину сгенерированной строки.
- B) Степень уникальности и разнообразия сгенерированных данных.
- C) Время обучения модели.
- D) Уровень риска переобучения.

Вопрос 18: Что характеризует высокую метричность Inception Score (IS)?

- A) Устойчивость модели к небольшим изменениям входных данных.
- B) Качество и четкость сгенерированных изображений.
- C) Длительность работы модели.
- D) Низкий объем потребляемой оперативной памяти.

Вопрос 19: Что такое RAG (Retrieval Augmented Generation)?

- A) Процесс обучения нейросети на увеличенном объеме данных.
- B) Метод, сочетающий извлечение информации и генеративные модели.
- C) Технология ускорения обучения с помощью параллельных вычислений.
- D) Новый подход к снижению потребления энергии нейросетями.

Вопрос 20: Какая область применения демонстрирует наибольший потенциал генеративных моделей?

- A) Медицина и биология.
- B) Искусство и дизайн.
- C) Финансовая сфера.
- D) Военная промышленность.

Вопрос 21: Какие факторы важны при создании качественного генеративного контента?

- A) Объем данных и наличие экспертных мнений.
- B) Большие объемы памяти компьютера.
- C) Наличие мощного оборудования.
- D) Качество данных и тщательная настройка гиперпараметров.

Вопрос 22: Какая главная трудность связана с созданием мульти-модальных генеративных моделей?

- A) Необходимость огромного объема данных.
- B) Сложность интеграции разнородных источников данных.
- C) Трудности в визуализации результата.
- D) Потребность в значительных финансовых вложениях.

Вопрос 23: Что определяет успех генеративной модели в современном мире?

- A) Скорость генерации данных.
- B) Креативный потенциал и новизна генерируемых данных.
- C) Экономичность и низкое потребление ресурсов.
- D) Способность избегать любых ошибок.

Вопрос 24: Какая модель преимущественно используется для синтеза реалистичных изображений высокого разрешения?

- A) Least Squares GAN (LS-GAN)
- B) Wasserstein GAN (WGAN)
- C) PixelRNN
- D) StyleGAN

Вопрос 25: Что характерно для высокоэффективных генеративных моделей?

- A) Очень медленный процесс обучения.
- B) Способность учиться на небольших наборах данных.
- C) Постоянное возникновение артефактов.
- D) Неустойчивость к любым изменениям входных данных.

Время выполнения: 30-40 минут.

Критерии оценки:

– оценку «отлично» получает студент, правильно ответивший на 82 и более процентов вопросов.

– оценку «хорошо» получает студент, правильно ответивший на 71-81 процентов вопросов.

– оценку «удовлетворительно» получает студент, правильно ответивший на 60-70 процентов вопросов.

– оценку «неудовлетворительно» получает студент, правильно ответивший менее, чем на 60 процентов вопросов.

Перечень компетенций (части компетенции), проверяемых оценочным средством:

LLM-1.1; LLM-1.7; FC-2.1; DL-1.1; DL-1.3; DL-2.1; DL-4.1.

Практические кейсы по тематике лабораторных работ

1. Основные концепции и типы генеративных моделей

Кейс №1: Генеративная сеть для восстановления поврежденных изображений

Описание задачи: Студенты получают фотографии известных российских памятников культуры (например, Эрмитаж, Третьяковская галерея), некоторые части которых повреждены (размытые участки, царапины, пятна). Необходимо разработать генеративную модель, способную восстановить оригинальные элементы изображения.

Этапы работы:

1. Загрузка и подготовка датасета изображений.
2. Реализация базовой генеративной модели (автокодировщик или вариационный автокодировщик).
3. Тестирование и оценка восстановленного изображения по субъективным критериям и метрикам.

2. Генеративные нейросети

Кейс №2: Генерация новых логотипов российских брендов

Описание задачи: Используя доступные публичные коллекции логотипов российских компаний, студенты разрабатывают модель, способную автоматически генерировать новый уникальный логотип для бренда на основе стилистической схожести с существующими примерами.

Этапы работы:

1. Сбор и очистка базы данных логотипов.
2. Проектирование и настройка архитектуры GAN или VAE для генерации логотипов.
3. Демонстрация лучших образцов и проверка их оригинальности.

Кейс №3: Диффузионная модель для воссоздания портретов исторических деятелей

Описание задачи: Необходимо создать модель, которая сможет постепенно добавлять детали к абстрактному наброску лица известного исторического деятеля России (например, Петра I, Александра Пушкина), создавая чёткое изображение.

Этапы работы:

1. Поиск и сбор данных лиц известных исторических персон.
2. Реализация диффузионной модели (diffusion model) для постепенного воссоздания деталей.
3. Проверка устойчивости модели и воспроизведения узнаваемых черт лица.

3. Мультимодальность и трансформеры

Кейс №4: Система автоматического написания новостей на основе фото

Описание задачи: Создать систему, которая способна автоматически формировать новостной заголовок и короткую заметку по предоставленному изображению (например, фотография политической акции, культурного мероприятия).

Этапы работы:

1. Формирование базы данных пар “изображение–заголовок”.
2. Реализация multimodal трансформера для генерации текста на основе изображения.
3. Оценка качества и точности полученной аннотации с помощью человеческих судей.

Кейс №5: Модель CLIP для выбора лучшей иллюстрации статьи

Описание задачи: Разработайте модель, которая выберет самую подходящую фотографию для размещения рядом с заданным текстом статьи на сайте.

Этапы работы:

1. Подготовка пары данных: статья и возможные иллюстрации.
2. Реализация модели CLIP или ruCLIP для сопоставления текста и изображений.
3. Подсчет процента успешных выборов иллюстраций среди случайных вариантов.

4. Технологии и приложения мультимодального генеративного ИИ

Кейс №6: Система перевода и озвучивания лекций на иностранный язык

Описание задачи: Студентам предлагается разработать приложение, которое способно перевести русский голос преподавателя лекции на английский язык и озвучить переведённый текст другим голосом.

Этапы работы:

1. Запись короткого фрагмента русской лекции преподавателем.
2. Использование Tacotron или аналогичных систем для превращения русского текста в английскую речь.
3. Обеспечение качественной синхронизации звучащей английской версии с оригиналом.

Кейс №7: Создание музыкального трека на основе введённого текста

Описание задачи: Требуется разработать модель, которая создаёт музыкальный трек в определённом стиле на основе литературного произведения (например, стихотворения Ахматовой или Есенина).

Этапы работы:

1. Выбор стиля музыки и соответствующего автора.
2. Реализация трансформера или специализированной модели для генерации мелодии.
3. Демонстрация музыкальных композиций и сравнение предпочтений слушателей.

Раздел 5. Методы оценки качества генеративных моделей

Кейс №8: Анализ генеративной модели красоты русских пейзажей

Описание задачи: Необходимо провести детальный анализ качества генеративной модели пейзажа (например, лес зимой или берег Волги летом), оценив с точки зрения метрик FID, IS и Perplexity.

Этапы работы:

1. Реализация генератора пейзажей (GAN, VAE или диффузионная модель).
2. Количественная оценка качества изображений по указанным метрикам.
3. Предоставление рекомендаций по улучшению модели на основе анализа результатов.

Раздел 6. Современные исследования и практические кейсы

Кейс №9: Создание чат-бота с элементами RAG

Описание задачи: Создать простого чат-бота, способного вести беседу на русском языке, интегрировав внешние источники информации для увеличения информативности ответов.

Этапы работы:

1. Установка и подключение готовой языковой модели (например, GPT-3).
2. Организация внешней базы знаний для извлечения дополнительной информации.
3. Оценка эффективности внедрения механизма RAG с помощью пользователей.

Пример лабораторной работы

Лабораторная работа №4. Диффузионные модели

Цель:

Изучив принцип работы диффузионных моделей, научиться генерировать высококачественные изображения с помощью популярной библиотеки *diffusers* от Hugging Face. Получить опыт в работе с гиперпараметрами и визуализации результатов генерации.

Задачи:

- Изучить концепцию диффузионных моделей и их отличие от классических генеративных моделей.
- Освоить работу с библиотекой *diffusers* и популярным решением *Stable Diffusion*.
- Реализовать генерацию изображений на основе текстовых запросов.
- Оценить влияние гиперпараметров (количество шагов, направленность шума) на результат генерации.
- Осуществить анализ полученных изображений и сделать выводы относительно качества.

Ожидаемые результаты:

- Понимание принципа работы диффузионных моделей.
- Способность генерировать разнообразные изображения с помощью Stable Diffusion.
- Владение основными параметрами настройки моделей и пониманием их влияния на результат.
- Возможность анализа и интерпретации полученного изображения, его соответствия запросу.

Инструменты и библиотеки:

- Язык программирования Python
- Google Colab или аналогичный сервис с возможностью использования GPU
- Библиотека diffusers от Hugging Face
- Пакеты для работы с изображениями (PIL, Matplotlib)

Исходные данные:

Будем использовать текстовые запросы и свободную генерацию изображений без привязки к конкретному датасету.

Ход работы:

1. Установка библиотек

Установка необходимых пакетов:

```
!pip install torch diffusers transformers accelerator bitsandbytes safetensors
```

2. Загрузка подготовленной модели

Выбор и загрузка предварительно обученной модели (например, Stable Diffusion v1.5):

```
from diffusers import StableDiffusionPipeline
```

```
model_id = "runwayml/stable-diffusion-v1-5"
pipe = StableDiffusionPipeline.from_pretrained(model_id, use_auth_token=True)
pipe.to("cuda") # Перемещаем модель на GPU
```

3. Генерация первых изображений

Простая генерация изображения на основе произвольного текстового запроса:

```
prompt = "Картина русской природы, зимний лес, заснеженная поляна"
image = pipe(prompt).images[0]
image.save("winter_forest.png")
```

4. Параметризация генерации

Испытайте различные параметры генерации, варьируя количество шагов и степень детализации:

```
prompt = "Городской пейзаж Москвы вечером"
image_20_steps = pipe(prompt, num_inference_steps=20).images[0]
image_100_steps = pipe(prompt, num_inference_steps=100).images[0]
```

```
image_20_steps.save("moscow_evening_20_steps.png")
image_100_steps.save("moscow_evening_100_steps.png")
```

5. Влияние степени шума

Исследуйте влияние параметра noise_level на резкость и детализацию изображения:

```
prompt = "Закат солнца над озером Байкал"
noise_levels = [0.2, 0.5, 0.8]
```

```
for level in noise_levels:
```

```
    image = pipe(prompt, guidance_scale=level).images[0]
    image.save(f"байкал_sunset_noise_{level:.2f}.png")
```

6. Совершенствование результатов

Повысьте качество изображений, настроив параметры модели и текстовые запросы:

```
prompt_advanced = "Художник написал картину красивого осеннего леса, яркие цвета листьев, мягкий свет солнца"
```

```
advanced_image = pipe(prompt_advanced, num_inference_steps=100,  
guidance_scale=7.5).images[0]
```

```
advanced_image.save("forest_autumn_advanced.png")
```

7. Анализ результатов

Проанализируйте влияние гиперпараметров на генерацию изображений, сделайте выводы о целесообразности тех или иных настроек.

Требования к отчету:

1. **Титульный лист.** Название работы, ФИО студента, номер группы, дата выполнения.
2. **Постановка задачи.** Четкая цель и перечень задач лабораторной работы.
3. **Краткий обзор теории.** Краткое описание диффузионных моделей и их отличия от классических генеративных моделей.
4. **Ход работы.** Показан весь код с подробными комментариями и описанием каждого шага.
5. **Результаты.** Демонстрация полученных изображений, сравнений и анализов эффектов различных параметров.
6. **Выводы.** Краткий вывод о проделанной работе, особенностях генерации изображений и рекомендациях по улучшению.
7. **Список литературы.** Все использованные ресурсы и учебные материалы.

Критерии оценки:

- **Отлично:** полное соблюдение требований, качественное исполнение всех пунктов, хорошо представленные и осмысленные результаты.
- **Хорошо:** правильное выполнение большинства задач, однако возможны незначительные ошибки или пропуски комментариев.
- **Удовлетворительно:** важные пункты пропущены, очевидные ошибки в результатах или недостаточная глубина анализа.
- **Неудовлетворительно:** невыполнение большей части заданий или серьезные ошибки в выполнении.

Проверяемые компетенции комплексом практических заданий: LLM-1.1; LLM-1.7; FC-2.1; DL-1.1; DL-1.3; DL-2.1; DL-4.1.

Зачетно-экзаменационные материалы для промежуточной аттестации (экзамен)

Вопросы для подготовки к экзамену

1. Чем отличаются генеративные модели от дискриминантных моделей?
2. Назовите основные классы генеративных моделей и приведите примеры.
3. Опишите общий подход к обучению генеративных моделей.
4. Что представляет собой задача восстановления изображения и латентное пространство?
5. Какие существуют стратегии уменьшения размерности признаков в генеративных моделях?
6. Почему важно понимать смысл и структуру данных при обучении генеративных моделей?

7. Какие популярные платформы и инструменты используются для создания и обучения генеративных моделей?
8. Что такое автокодировщик (Autoencoder)? Какова его структура и назначение?
9. Чем отличается обычный автокодировщик от вариационного (VAE)?
10. Какие бывают функции активации в вариационном автокодировщике и как они влияют на результаты?
11. Опишите основной принцип работы диффузионных моделей (Diffusion Models).
12. Чем различаются стабильные диффузионные модели (Stable Diffusion) и латентные диффузионные модели (Latent Diffusion)?
13. В чём заключается идея состязательных генеративных сетей (GANs)? Приведите известные архитектуры GAN.
14. Что означает термин «Expert-in-the-loop» (MoE) и как эта концепция применяется в генеративных сетях?
15. Что такое мультимодальность в контексте генеративных моделей?
16. Опишите общую архитектуру мультиспектрального трансформера (Multimodal Transformer).
17. Как механизмы внимания (attention mechanisms) помогают обрабатывать мультиспектральные данные?
18. Что такое технология image captioning и как она реализуется на практике?
19. Как работает cross-modal learning и какие задачи оно решает?
20. Опишите модели типа CLIP и ruCLIP. В чём их сходство и различие?
21. В чём особенность интерактивных нейроархитектур типа Flamingo?
22. Где хранятся готовые генеративные модели и как их загружают для своей задачи?
23. Как выбрать оптимальные гиперпараметры (batch size, sample rate) для конкретной задачи?
24. Какие задачи решают модели для распознавания и генерации речи (например, Whisper, Wav2Vec, Tacotron)?
25. Приведите достоинства и ограничения перечисленных моделей для задач обработки речи.
26. Приведите пример сценария совместного использования нескольких модулей (видео, аудио, текст) в одном проекте.
27. Какие актуальные коммерческие продукты используют генеративные модели?
28. Что показывает метрика Maximum Mean Discrepancy (MMD)? Как её рассчитать?
29. В чём состоит суть метрики Frechet Inception Distance (FID)? Когда её применяют?
30. Что обозначает Inception Score (IS)? Какая связь между IS и восприятием качества человеком?
31. Какой смысл вкладывается в понятие Perplexity и как оно применяется в задачах NLP?
32. Что измеряет метрику Diversity и зачем она важна?
33. Как выбираются оптимальные метрики для конкретного класса задач?
34. Приведите примеры использования RAG (Retrieval-Augmented Generation) в коммерческой практике.
35. Что представляют собой комплексные сценарии совместных генеративных задач (например, совместное производство аудио и видеоматериалов)?
36. Опишите тенденцию развития индустрии генеративного ИИ и ожидаемые направления исследований.
37. Какие перспективы имеют генеративные модели в будущем?
38. Приведите два-три примера успешного применения генеративных моделей в бизнесе.
39. Какие главные вызовы стоят перед отраслью генеративного ИИ на сегодняшний день?

Перечень компетенций (части компетенции), проверяемых оценочным средством
LLM-1.1; LLM-1.7; FC-2.1; DL-1.1; DL-1.3; DL-2.1; DL-4.1.

Практические задания к экзамену

Задание 1. Генеративные нейросети: Автоэнкодеры (Autoencoders)

Условия: Используя набор данных MNIST, создайте автоэнкодер, который восстанавливает изображения рукописных цифр.

Задачи:

1. Реализуйте однокомпонентный автоэнкодер с линейной структурой (input → hidden → output).
2. Постройте модель с нелинейной активацией (например, Relu) и скрытым слоем меньшей размерности.
3. Обучите обе модели и сравните результаты реконструкции изображений.

Ожидаемый результат: Два обученных автоэнкодера, демонстрация разницы между реконструкцией линейной и нелинейной моделей.

Задание 2. Вариационные автокодировщики (VAE)

Условия: Используя тот же набор данных MNIST, постройте вариационный автокодировщик (VAE).

Задачи:

1. Реализуйте структуру VAE с одним скрытым слоем.
2. Обучите модель и продемонстрируйте реконструкцию исходных изображений.
3. Исследуйте поведение латентного пространства, генерируя изображения с изменением параметров в пространстве Z.

Ожидаемый результат: Модель VAE, продемонстрированная реконструкция и исследование свойств латентного пространства.

Задание 3. Диффузионные модели (Diffusion Models)

Условия: Используя библиотеку diffusers, создайте и запустите генеративную модель для синтеза изображений.

Задачи:

1. Выберите одну из популярных моделей (например, Stable Diffusion).
2. Сгенерируйте изображение на основе текстового запроса.
3. Изучите влияние числа шагов генерации и размера выборки на качество результата.

Ожидаемый результат: Сгенерированное изображение высокого качества, исследовательская работа по изменению гиперпараметров.

Задание 4. Генеративно-сопоставительные сети (GANs)

Условия: Используя базу данных CIFAR-10, постройте простую генеративно-сопоставительную сеть (GAN).

Задачи:

1. Реализуйте архитектуру GAN (generator и discriminator).
2. Обучите модель на CIFAR-10 и покажите сгенерированные изображения.
3. Проверьте устойчивость и производительность модели на небольшом количестве эпох.

Ожидаемый результат: Работаящий GAN, коллекция сгенерированных изображений и комментарий о стабильности обучения.

Задание 5. Multimodal Transformers: Image Captioning

Условия: Используя популярный датасет COCO, создайте модель для генерации подписей к изображениям.

Задачи:

1. Подготовьте данные из СОСО для обучения.
2. Постройте трансформерную модель, объединяя обработку изображения и текста.
3. Оцените качество генерации подписи, проведя тестирование на валидных данных.

Ожидаемый результат: Модифицированный трансформер, способный генерировать текстовые подписи к изображениям.

Задание 6. Метрики качества генеративных моделей

Условия: Используя собственный или готовый набор данных, рассчитайте метрики качества для вашей генеративной модели.

Задачи:

1. Вычислите FID-метрику для набора синтетических изображений.
2. Найдите Inception Score (IS) и обсудите его значимость.
3. Используйте метрику Perplexity для оценки текстовой модели.

Ожидаемый результат: Показатели метрик, сопровождающиеся анализом их значимости и выводами о качестве модели.

Задание 7. Решение комплексной задачи генеративного ИИ

Условия: Разработайте решение, включающее сразу несколько модулей генеративного ИИ (например, speech-to-text и text-to-image).

Задачи:

1. Объедините модули для последовательного преобразования данных (например, запись аудио → расшифровка текста → генерация изображения).
2. Проверьте работоспособность всего конвейера на примере нескольких вводных данных.
3. Обсудите возможности расширения функционала и улучшения качества.

Ожидаемый результат: Комплексное решение, состоящее из нескольких взаимосвязанных компонентов генеративного ИИ.

Перечень компетенций (части компетенции), проверяемых оценочным средством LLM-1.1; LLM-1.7; FC-2.1; DL-1.1; DL-1.3; DL-2.1; DL-4.1.

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

Методические рекомендации, определяющие процедуры оценивания на экзамене:

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

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

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

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

Форма проведения экзамена: устно.

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

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

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

Критерии оценки:

1. Оценка ответов на экзаменационные вопросы (40% итоговой оценки)

Отлично (5)

Полные, развернутые ответы с демонстрацией глубокого понимания темы.
Использование примеров, формул, корректных терминов.
Умение анализировать и сравнивать (например, функции активации или оптимизаторы).

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

Хорошо (4)

Ответы содержат основные идеи, но без углубленного анализа.
Возможны небольшие ошибки в деталях или формулировках.

% выполнения: 75–89%.

Удовлетворительно (3)

Ответы поверхностные, с существенными пробелами.
Отсутствие примеров или некорректное применение терминов.

% выполнения: 60–74%.

Неудовлетворительно (2)

Отсутствие понимания ключевых концепций.
Грубые ошибки или неспособность ответить на большую часть вопросов.

% выполнения: <60%.

2. Оценка выполнения практических кейсов и лабораторных работ (40% итоговой оценки)

Отлично (5)

Полное выполнение всех этапов кейса с инновационными решениями.
Достижение целевых показателей обучения.
Четкая документация кода и анализ результатов.

% выполнения: 90–100%.

Хорошо (4)

Выполнены основные задачи, но без дополнительной оптимизации.
Незначительные отклонения от целевых показателей обучения (например, ошибка больше желаемой на 3-5%).

% выполнения: 75–89%.

Удовлетворительно (3)

Решены базовые задачи, но с критическими ошибками.
Низкое качество кода или отсутствие анализа.

% выполнения: 60–74%.

Неудовлетворительно (2)

Невыполнение ключевых этапов.
Код нерабочий или отсутствует.

% выполнения: <60%.

3. Оценка тестовых вопросов (20% итоговой оценки)

Отлично (5)

24–28 правильных ответов (82–100%).

Демонстрация уверенного владения терминологией и методами.

Хорошо (4)

20–23 правильных ответов (71–81%).

Незначительные ошибки в сложных вопросах.

Удовлетворительно (3)

16–19 правильных ответов (60–70%).

Путаница в базовых концепциях (например, путает назначение архитектур или не понимает разницу в оптимизаторах).

Неудовлетворительно (2)

Менее 16 правильных ответов (<60%).

Неспособность отличить архитектуры (например, многослойный персептрон от спайковой сети).

Итоговая оценка (суммарно)

| Оценка | Экзамен (40%) | Практика (40%) | Тест (20%) | Общий % |
|-------------------------|---------------|----------------|------------|---------|
| Отлично (5) | 90–100% | 90–100% | 82–100% | ≥85% |
| Хорошо (4) | 75–89% | 75–89% | 71–81% | 72–85% |
| Удовлетворительно (3) | 60–74% | 60–74% | 60–70% | 60–71% |
| Неудовлетворительно (2) | <60% | <60% | <60% | <60% |

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

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

Практические кейсы оцениваются по:

- Корректности кода.
- Достижению целевых показателей обучения.
- Качеству отчета (анализ ошибок, визуализация).

Методические рекомендации, определяющие процедуры оценивания лабораторных работ:

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

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

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

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

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

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

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

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

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

Для лиц с нарушениями зрения:

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

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4.3. Методические указания по организации вычислительной инфраструктуры

Условия применения:

- Курс рассчитан на студентов 4-го года обучения.
- Наличие доступа к вычислительным ресурсам (GitLab, Google Colab или Yandex DataSphere, JupyterHub, Hugging Face).
- Разработаны лабораторные работы;
- Инфраструктура для приёма задач (gitlab, CI/CD) согласована с лабораторными работами.

Цели, задачи и ожидаемые результаты

Цели организации вычислительной инфраструктуры:

- дать представление о работе в IT инфраструктуре (приучить пользоваться гитом, jupyter-ноутбуками).

Задачи преподавателя:

- Организация регистрации студентов в Google Colab и Yandex DataSphere
- Настройка GitLab Runner для автоматического тестирования кода.
- Разработка шаблонного репозитория для лабораторных работ с предустановленными зависимостями (PyTorch, Hugging Face Transformers, Hugging Face Diffusers, Optuna).
- Написание автотестов для проверки корректности выполнения заданий (например, fine-tuning моделей).
- Визуализация результатов тестирования через HTML-отчеты.
- Подготовка инструкций по работе с Git и облачными ресурсами.

Ожидаемые результаты студентов:

- представление о работе в IT инфраструктуре (гит, нейминг).
- Навыки запуска и тестирования генеративных нейросетей в облачных средах.
- Понимание CI/CD-процессов в контексте разработки генеративного ИИ.

Порядок реализации

Задача №1: Организация регистрации студентов в Google Colab и Yandex DataSphere

Задача №2: Настройка GitLab Runner:

Для автоматического тестирования кода используется Docker-образ с предустановленными библиотеками (PyTorch, Hugging Face Transformers, Hugging Face Diffusers, Optuna).

Для выполнения CI/CD пайплайна был настроен GitLab Runner на удалённой виртуальной машине с ОС Ubuntu 24.04.

Последовательность настройки включала следующие шаги:

- Настройка системы – установка необходимых компонентов, таких, как Docker.
- Установка GitLab Runner по официальной инструкции.
- Регистрация Runner для частного сервера GitLab.

Задача №3: Шаблонный репозиторий:

Включает:

.gitlab-ci.yml для CI/CD.

Скрипты для предобработки данных и обучения моделей.

Примеры кода для работы с GAN-сетями и другими архитектурами.

Задача №4: Автотесты:

Проверяют корректность разработанных генеративных нейросетей.

Задача №5: Визуализация результатов:

Генерация HTML-отчетов с результатами тестирования, включая метрики качества моделей.

Порядок проверки корректности:

- Наличие Git-репозитория у всех студентов.
- Шаблонный репозиторий с подключенными автотестами.
- Инструкция по работе с Git и CI/CD в формате README.md.

Вся структура максимально адаптирована для копирования студентами и минимизации порога входа при выполнении лабораторных

4.4. Методические указания по организации лабораторных работ

Условия применения:

- Курс рассчитан на студентов 4-го года обучения.
- Наличие доступа к вычислительным ресурсам (GitLab) и к GPU/CPU (Kaggle, локальные серверы).
- Разработана инфраструктура для приёма задач (Gitlab, CI/CD) и согласована с лабораторными работами и настроена на всех студентов образовательной программы;
- Использование открытых датасетов и библиотек.

Цели, задачи и ожидаемые результаты

Цели организации лабораторных работ:

- Закрепление теоретических знаний на практике.
- Развитие навыков разработки глубоких генеративных нейронных сетей.
- Подготовка к решению реальных задач в индустрии.

Задачи преподавателя:

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

Ожидаемые результаты студентов:

- Умение применять глубокие генеративные нейронные сети на практике.
- Владение библиотеками и фреймворками (PyTorch, Hugging Face Transformers, Hugging Face Diffusers, Optuna).
- Опыт работы с мультимодальными моделями.
- Опыт работы с генерацией изображений, текста и аудио.

Порядок реализации

Задача №1: Подготовка лабораторных работ (в соответствии с п. 2.3.3 РПД)

1) Определение тем:

- Введение в генеративные модели
- Работа с автокодировщиками (Autoencoders)
- Реализация вариационного автокодировщика (VAE)
- Диффузионные модели (Diffusion Models)
- Генеративно-сопоставительные сети (GANs)
- Introduction to Multimodal Data Processing
- Реализация Multimodal Transformers
- Генераторы текста и речи
- Гиперпараметризация моделей
- Анализ метрик генеративных моделей
- Case Study in Retrieval-Augmented Generation (RAG)

| Наименование Лабораторной работы | Содержание Лабораторной работы | Распределение часов |
|--|--|------------------------------|
| 2 | 3 | |
| Введение в генеративные модели | Классификация генеративных моделей (пиксельные, вариационные, сопоставительные, диффузионные). Принципы работы генеративных моделей. Основные показатели эффективности (log likelihood, FID score, SSIM, etc.). Инструменты и среды для моделирования (Python, PyTorch). Практическое задание: Создать простую генеративную модель, использующую распределение Гаусса для генерации случайных чисел. | 3 часа (2 часа ЛР, 1 час СР) |
| Работа с автокодировщиками (Autoencoders) | Основная структура автокодировщика: энкодер и декодер. Потеря информации и способы минимизации потерь. Алгоритм обучения автокодировщика. Применение автокодировщиков для компрессии изображений. Практическое задание: Реализовать простейший автокодировщик на MNIST датасете. | 3 часа (2 часа ЛР, 1 час СР) |
| Реализация вариационного автокодировщика (VAE) | Основные отличия VAE от классического автокодировщика. Вероятностная интерпретация VAE. Формула цели обучения (ELBO). Использование VAE для генерации новых данных. Практическое задание: Реализовать VAE на примере MNIST или CelebA датасета. | 4 часа (3 часа ЛР, 1 час СР) |
| Диффузионные модели (Diffusion Models) | Изучив принцип работы диффузионных моделей, научиться генерировать высококачественные изображения с помощью популярной библиотеки diffusers от Hugging Face. Получить опыт в работе с | 3 часа (2 часа ЛР, 1 час СР) |

| Наименование Лабораторной работы | Содержание Лабораторной работы | Распределе ние часов |
|--|---|------------------------------|
| 2 | 3 | |
| | гиперпараметрами и визуализации результатов генерации. | |
| Генеративно-состязательные сети (GANs) | Концепция GAN: генератор и дискриминатор. Типичные архитектуры GAN (DCGAN, StyleGAN). Проблемы обучения GAN (нестабильность, коллапс режима). Примеры успешного применения GAN. Практическое задание: Реализация DCGAN на датасете лиц (CelebA) или картинок животных. | 4 часа (3 часа ЛР, 1 час СР) |
| Introduction to Multimodal Data Processing | Понятие мультимодальности. Примеры мультимодальных: задача совместимости текста и изображения. Артефакты синхронизации и проблема измерения сходства. Техники объединения признаков и методы оценивания. Практическое задание: Простая модель совместного представления изображений и подписей к ним. | 4 часа (3 часа ЛР, 1 час СР) |
| Реализация Multimodal Transformers | Особенности работы с разными модальностями в одной архитектуре. Пример архитектуры CLIP (Contrastive Language-Image Pre-training). Использование больших pretrained-трансформеров (BERT, ViT). Оценка эффективности мультитрансформеров. Практическое задание: Модификация архитектуры CLIP для генерации подписей к картинкам. | 4 часа (3 часа ЛР, 1 час СР) |
| Генераторы текста и речи | Основные компоненты языковой модели: embedding layer, hidden layers, output layer. Рекуррентные нейронные сети (RNN) и их ограничения. Механизмы запоминания контекста: LSTM и GRU. Автокодировщики и последовательность-последовательность (Seq2Seq)-модели. Использование трансформеров (Transformer architecture) для генерации текста. Fine-tuning готовых моделей (GPT, BERT, RoBERTa). Валидация и метрические оценки: BLEU, ROUGE, perplexity. Практическое задание: Реализация простой генеративной модели для перевода коротких сообщений с русского на английский, используя seq2seq-модель с механизмом внимания (attention mechanism). Задача: Реализация нейро-символьной системы для обработки медицинских данных пациентов с | 4 часа (3 часа ЛР, 1 час СР) |

| Наименование Лабораторной работы | Содержание Лабораторной работы | Распределение часов |
|--|--|------------------------------|
| 2 | 3 | |
| | помощью фреймворков DeepProbLog/Neurosymbolic AI Toolkit | |
| Гиперпараметризация моделей | <p>Значение гиперпараметров в обучении генеративных моделей.</p> <p>Классические методы перебора гиперпараметров: grid search, random search.</p> <p>Продвинутые методы оптимизации: байесовская оптимизация, эволюционная стратегия.</p> <p>Программные инструменты для автоматизации поиска гиперпараметров: Hyperopt, Optuna, Ray Tune.</p> <p>Оптимизация гиперпараметров на примерах генеративных моделей (GAN, VAE, трансформеры).</p> <p>Оценка влияния различных значений гиперпараметров на скорость обучения и качество генерации.</p> <p>Практическое задание: Используя библиотеку Optuna, провести оптимизацию гиперпараметров генеративной модели Variational Autoencoder (VAE) для максимального увеличения показателя log-likelihood на заданном датасете.</p> | 4 часа (3 часа ЛР, 1 час СР) |
| Анализ метрик генеративных моделей | <p>Frechet Inception Distance (FID).</p> <p>Structural Similarity Index Measure (SSIM).</p> <p>Pixel-to-pixel различия.</p> <p>Плюсы и минусы различных метрик.</p> <p>Практическое задание: Вычислить метрики FID и SSIM для набора изображений, созданных моделью VAE и реальной выборки.</p> | 3 часа (2 часа ЛР, 1 час СР) |
| Case Study in Retrieval-Augmented Generation (RAG) | <p>Concept of retrieval-augmented generation.</p> <p>Модификации трансформеров с добавлением внешнего знания.</p> <p>Обзор популярных моделей: RAG-T5, FiD.</p> <p>Методы интеграции сторонних данных в генеративные модели.</p> <p>Практическое задание: Прототип RAG-модели для дополнения диалогов или текста справочной</p> | 3 часа (2 часа ЛР, 1 час СР) |

2) Разработка заданий:

- Пошаговые инструкции.
- Примеры кода.

Контрольные вопросы.

Разработка заданий для лабораторной работы «Диффузионные модели»

Цель:

Изучив принцип работы диффузионных моделей, научиться генерировать высококачественные изображения с помощью популярной библиотеки diffusers от Hugging Face. Получить опыт в работе с гиперпараметрами и визуализации результатов генерации.

2. Пошаговые инструкции

Будем использовать текстовые запросы и свободную генерацию изображений без привязки к конкретному датасету.

Шаг 1: Установка библиотек

Шаг 2: Загрузка подготовленной модели

Шаг 3: Генерация первых изображений

Шаг 4: Параметризация генерации

Шаг 5. Влияние степени шума

Шаг 6. Совершенствование результатов

Шаг 7. Анализ результатов

3. Примеры кода на Python

Полный скрипт

Ход работы:

1. Установка библиотек

Установка необходимых пакетов:

```
!pip install torch diffusers transformers accelerator bitsandbytes safetensors
```

2. Загрузка подготовленной модели

Выбор и загрузка предварительно обученной модели (например, Stable Diffusion v1.5):

```
from diffusers import StableDiffusionPipeline
```

```
model_id = "runwayml/stable-diffusion-v1-5"
```

```
pipe = StableDiffusionPipeline.from_pretrained(model_id, use_auth_token=True)
```

```
pipe.to("cuda") # Перемещаем модель на GPU
```

3. Генерация первых изображений

Простая генерация изображения на основе произвольного текстового запроса:

```
prompt = "Картина русской природы, зимний лес, заснеженная поляна"
```

```
image = pipe(prompt).images[0]
```

```
image.save("winter_forest.png")
```

4. Параметризация генерации

Испытайте различные параметры генерации, варьируя количество шагов и степень детализации:

```
prompt = "Городской пейзаж Москвы вечером"
```

```
image_20_steps = pipe(prompt, num_inference_steps=20).images[0]
```

```
image_100_steps = pipe(prompt, num_inference_steps=100).images[0]
```

```
image_20_steps.save("moscow_evening_20_steps.png")
```

```
image_100_steps.save("moscow_evening_100_steps.png")
```

5. Влияние степени шума

Исследуйте влияние параметра `noise_level` на резкость и детализацию изображения:

```
prompt = "Закат солнца над озером Байкал"
```

```
noise_levels = [0.2, 0.5, 0.8]
```

```
for level in noise_levels:
```

```
    image = pipe(prompt, guidance_scale=level).images[0]
```

```
    image.save(f"байкал_sunset_noise_{level:.2f}.png")
```

6. Совершенствование результатов

Повысьте качество изображений, настроив параметры модели и текстовые запросы:

```
prompt_advanced = "Художник написал картину красивого осеннего леса, яркие цвета листьев, мягкий свет солнца"
advanced_image = pipe(prompt_advanced, num_inference_steps=100,
guidance_scale=7.5).images[0]
advanced_image.save("forest_autumn_advanced.png")
```

7. Анализ результатов

Проанализируйте влияние гиперпараметров на генерацию изображений, сделайте выводы о целесообразности тех или иных настроек.

4. Контрольные вопросы

1. Теоретические:
 - Что представляет собой процесс диффузии в рамках работы генеративных моделей?
 - Какие преимущества имеют диффузионные модели перед традиционными моделями генерации изображений?
 - Почему шум играет важную роль в процессе генерации изображений?
2. Практические:
 - Опишите шаги установки библиотеки `diffusers` и загрузки предобученных моделей.
 - Напишите простой скрипт для генерации изображения по текстовому запросу с указанием числа шагов генерации.
 - Объясните разницу между результатами генерации при различном уровне шума (`guidance scale`).
3. Аналитические:
 - Какое влияние оказывает увеличение числа шагов генерации на конечный результат?
 - Какой оптимальный уровень шума выбрать для получения наилучших результатов?
 - Какие факторы влияют на общее качество генерации изображений с помощью диффузионных моделей?

Критерии оценки

Отлично: полное соблюдение требований, качественное исполнение всех пунктов, хорошо представленные и осмысленные результаты.

Хорошо: правильное выполнение большинства задач, однако возможны незначительные ошибки или пропуски комментариев.

Удовлетворительно: важные пункты пропущены, очевидные ошибки в результатах или недостаточная глубина анализа.

Неудовлетворительно: невыполнение большей части заданий или серьезные ошибки в выполнении.

3) Подготовка датасетов:

- Использование открытых данных Kaggle, paperswithcode и Hugging Face.
- Генерация синтетических данных при необходимости.

Задача №2: доступ к необходимым вычислительным ресурсам (в п.4.3 РПД)

Задача №3: Организация проверки и обратной связи по выполненным работам.

Порядок проверки корректности

Чек-лист для проверки лабораторных работ:

1. Выполнение заданий:
 - Код запускается без ошибок.
 - Достигнуто целевое качество модели.
2. Качество кода:
 - Соблюдение PEP-8.

Наличие комментариев.

3. Отчет:

инструкция по работе с гитом с подробным описанием именования методов и коммитов;

Описание хода работы.

Анализ результатов.

4. Своевременность:

Работа сдана в установленный срок.

Критерии оценки:

Отлично: Полное выполнение всех заданий, качественный код и отчет.

Хорошо: Незначительные недочеты в коде или отчете.

Удовлетворительно: Выполнены базовые задания, но с ошибками.

Неудовлетворительно: Критические ошибки или невыполнение работы.

4.5. Методические указания по организации проектной деятельности студентов

Условия применения:

Курс рассчитан на студентов 4-го года обучения,

Общее время на проект – не более 16 часов на каждого студента.

Имеется доступ к кейсам индустриальных партнеров.

Цели, задачи и ожидаемые результаты

Цели организации вычислительной инфраструктуры:

дать начальное представление о реальных задачах, решаемых с помощью генеративного искусственного интеллекта и возникающих проблемах.

Задачи преподавателя:

- сбор кейсов индустриальных партнеров;
- сбор кейсов преподавателей практиков и лабораторий в вузе;
- формирование ТЗ на экзаменационный проект на основе кейсов;
- разработка системы учёта результатов проекта в итоговой оценке экзамена

Ожидаемые результаты студентов:

начальное представление о реальных задачах, решаемых с помощью генеративного ИИ и возникающих проблемах.

Порядок реализации

Задача №1: сбор кейсов индустриальных партнеров

1. Генеративный ИИ для автоматического составления инвестиционных обзоров

Описание: Аналитики Сбера ежедневно составляют десятки аналитических и инвестиционных обзоров по рынкам, компаниям, макроэкономике. Задача — исследовать применение LLM для генерации кратких сводок и аналитических отчетов на основе входных данных: биржевые котировки, макроэкономические показатели, рыночные события.

Цель: Разработать инструмент, способный по структурированным данным и краткому описанию формировать инвестиционный обзор в деловом стиле.

Ожидаемый результат: Модель, генерирующая аналитические тексты длиной 500–1000 слов с разделами «обзор событий», «рекомендации», «прогнозы», оформленные в формате банка.

2. Генеративный ИИ для создания проектной документации по ТЗ

Описание: В рамках проектирования объектов девелоперской компании архитекторы и инженеры тратят значительное время на подготовку текстовой проектной документации (обоснование решений, пояснительные записки, описания инженерных систем). Задача — исследовать возможность использования LLM для генерации черновиков проектной документации на основе исходных данных: этажность, материалы, климат, назначение, нормы.

Цель: Разработать прототип текстового генератора, который помогает специалистам быстрее формировать документацию в соответствии с шаблонами и нормативами.

Ожидаемый результат: Инструмент на основе LLM, создающий логически стройный и нормативно грамотный текст, поддающийся быстрой правке инженером.

Задача № 2: кейсов преподавателей практиков и лабораторий в вузе.

1. Генеративная сеть для восстановления поврежденных изображений

Описание задачи: Студенты получают фотографии известных российских памятников культуры (например, Эрмитаж, Третьяковская галерея), некоторые части которых повреждены (размытые участки, царапины, пятна). Необходимо разработать генеративную модель, способную восстановить оригинальные элементы изображения.

Этапы работы:

1. Загрузка и подготовка датасета изображений.
2. Реализация базовой генеративной модели (автокодировщик или вариационный автокодировщик).
3. Тестирование и оценка восстановленного изображения по субъективным критериям и метрикам.

2. Генерация новых логотипов российских брендов

Описание задачи: Используя доступные публичные коллекции логотипов российских компаний, студенты разрабатывают модель, способную автоматически генерировать новый уникальный логотип для бренда на основе стилистической схожести с существующими примерами.

Этапы работы:

1. Сбор и очистка базы данных логотипов.
2. Проектирование и настройка архитектуры GAN или VAE для генерации логотипов.
3. Демонстрация лучших образцов и проверка их оригинальности.

Задача №3: формирование ТЗ на экзаменационный проект на основе кейсов

1. Attention Mechanisms

Используя популярный датасет COCO, создайте модель для генерации подписей к изображениям. Проект выполняется в командах от 1 до 3 человек. Оценивается вся команда одной оценкой.

Индивидуальное задание состоит в анализе качества разработанных нейросетей. Для выполнения задания необходимо выполнить несколько задач:

1. подготовить набор данных для проведения тестирования;
2. протестировать разработанные нейросети,
3. посчитать функцию потерь
4. построить ошибки на обучающих и тестовых данных.
5. посчитать метрики
6. сделать выводы о качестве нейросети

В зависимости от качества теоретического ответа и количества реализованного самостоятельно кода преподаватель выставляет оценку.

Критерии оценки:

Отлично: Полное выполнение всех шагов, анализ полученных результатов с построением функции потерь и необходимых метрик, ответил на все теоретические вопросы.

Хорошо: Полное выполнение всех шагов, анализ полученных результатов с построением функции потерь и необходимых метрик, ответил не на все теоретические вопросы.

Удовлетворительно: Выполнены не все шаги, анализ полученных результатов выполнен не полностью, ответил не на все теоретические вопросы.

Неудовлетворительно: Невыполнение ключевых этапов. Не ответил на теоретические вопросы.

Оценку можно повысить, реализовав требуемый функционал или ответив дополнительно или заново на необходимые вопросы.

Требования для повышения оценки и итоговую оценку формирует преподаватель.

Задача №4: разработка системы учёта результатов проекта в итоговой отметке за экзамен

Выполнено в РПД, п 4.2

Порядок проверки корректности

Чек-лист для проверки лабораторных работ:

Набор кейсов индустриальных партнеров – 20 шт;

Набор кейсов преподавателей практиков и лабораторий ВУЗа – 9 шт;

Набор ТЗ в количестве 7 штук.

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

5.1 Основная литература:

1. Хливненко, Л. В. Практика нейросетевого моделирования : учебное пособие для вузов / Л. В. Хливненко, Ф. А. Пятакович. — 4-е изд., стер. — Санкт-Петербург : Лань, 2024. — 200 с. — ISBN 978-5-507-47590-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/393482> (дата обращения: 21.07.2025). — Режим доступа: для авториз. пользователей.

2. Ростовцев, В. С. Искусственные нейронные сети : учебник для вузов / В. С. Ростовцев. — 5-е изд., стер. — Санкт-Петербург : Лань, 2025. — 216 с. — ISBN 978-5-507-50568-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/447392> (дата обращения: 21.07.2025). — Режим доступа: для авториз. пользователей.

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).

5.2 Дополнительная литература:

1. 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

2. 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

3. 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>.

4. Kadurin, Artur, et al. "The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology." Oncotarget 8.7 (2016): 10883.

5. 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 pharmaceuticals 14.9 (2017): 3098-3104.

6. Polykovskiy, Daniil, et al. "Molecular sets (MOSES): a benchmarking platform for molecular generation models." Frontiers in pharmacology 11 (2020): 565644.

7. Khrabrov, Kuzma, et al. "\$\nabla^2\$ DFT: A Universal Quantum Chemistry Dataset of Drug-Like Molecules and a Benchmark for Neural Network Potentials." *Advances in Neural Information Processing Systems* 37 (2024): 36869-36889.
8. Polykovskiy, Daniil, et al. "Entangled conditional adversarial autoencoder for de novo drug discovery." *Molecular pharmaceutics* 15.10 (2018): 4398-4405.
9. Николенко, Сергей, Кадури, Артур и Архангельская Екатерина. Глубокое обучение. "Издательский дом"" Питер""", 2017.
10. Митяков, Е. С. Искусственный интеллект и машинное обучение : учебное пособие для вузов / Е. С. Митяков, А. Г. Шмелева, А. И. Ладынин. — Санкт-Петербург : Лань, 2025. — 252 с. — ISBN 978-5-507-51465-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/450827> (дата обращения: 21.07.2025). — Режим доступа: для авториз. пользователей.
11. 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).
12. 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).
13. Златопольский Д. М. Основы программирования на языке Python. 2-е изд. Москва: ДМК Пресс, 2018.

5.3. Периодические издания:

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

Конференции А*:

1. <https://openreview.net/forum?id=FMMF1a9ifL>
2. <https://openreview.net/forum?id=EIUrNM9U8c#discussion>
3. <https://openreview.net/forum?id=JoO6mtCLHD>
4. <https://aclanthology.org/2024.findings-emnlp.760/>
5. <https://aclanthology.org/2020.coling-main.588/>
6. https://link.springer.com/chapter/10.1007/978-3-030-72113-8_30
7. https://link.springer.com/chapter/10.1007/978-3-031-42448-9_10
8. <https://aclanthology.org/2024.findings-naacl.288/>

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.prlib.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: <https://experiments.springernature.com/sources/springer-protocols>
13. Springer Materials: <http://materials.springer.com/>
14. Nano Database: <https://nano.nature.com/>
15. Springer eBooks (i.e. 2020 eBook collections): <https://link.springer.com/>
16. "Лекториум ТВ" <http://www.lektorium.tv/>
17. 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 и C++
3. Google Scholar/arXiv – доступ к научным публикациям

Ресурсы свободного доступа

1. Официальная документация по библиотеке numpy <https://numpy.org/doc/>
2. Официальная документация по библиотеке pandas <https://pandas.pydata.org/docs/>
3. Официальная документация по библиотеке matplotlib <https://matplotlib.org/stable/index.html>.
4. Официальная документация по библиотеке scikit-learn <https://scikit-learn.org/stable/>
5. Официальная документация по библиотеке seaborn <https://seaborn.pydata.org/>
6. КиберЛенинка <http://cyberleninka.ru/>;
7. Американская патентная база данных <http://www.uspto.gov/patft/>
8. Министерство науки и высшего образования Российской Федерации <https://www.minobrnauki.gov.ru/>;
9. Федеральный портал "Российское образование" <http://www.edu.ru/>;
10. Информационная система "Единое окно доступа к образовательным ресурсам" <http://window.edu.ru/>;
11. Единая коллекция цифровых образовательных ресурсов <http://school-collection.edu.ru/> .
12. Проект Государственного института русского языка имени А.С. Пушкина "Образование на русском" <https://pushkininstitute.ru/>;
13. Справочно-информационный портал "Русский язык" <http://gramota.ru/>;
14. Служба тематических толковых словарей <http://www.glossary.ru/>;
15. Словари и энциклопедии <http://dic.academic.ru/>;
16. Образовательный портал "Учеба" <http://www.uceba.com/>;
17. Законопроект "Об образовании в Российской Федерации". Вопросы и ответы http://xn--273--84d1f.xn--plai/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 «Генеративный искусственный интеллект» предусмотрено проведение лекционных занятий, на которых даётся систематизированное представление об основах генеративных моделей. Курс посвящен изучению принципов работы, устройства и методов обучения генеративных моделей искусственного интеллекта. В ходе лекций студенты знакомятся с историей возникновения и развитием генеративных моделей, изучают основные архитектуры (автоэнкодеры, вариационные автокодировщики, диффузионные модели, генеративно-состязательные сети и т.д), а также демонстрируются практические примеры реализации генеративных моделей в задачах синтеза изображений, обработки мультимодальных данных, генерации текста и речи. Важным разделом курса является обучение студентов методам оценки качества генеративных моделей.

Лабораторные занятия курса посвящены практическому освоению разработки генеративных моделей различного типа и оценке качества их работы. Студенты на занятиях используют готовые решения из популярных библиотек (Hugging Face, diffusers и др.), работают с реальными и синтетическими датасетами, свободно доступными в открытом доступе, применяя библиотеки Python (numpy, pandas, matplotlib, seaborn и др.). После каждого лабораторного занятия даются задания для самостоятельного закрепления материала, направленные на проработку отдельных этапов разработки генеративных моделей или модификацию рассмотренных алгоритмов.

В рамках самостоятельной работы студенты обязаны углубленно изучать рекомендованную литературу (учебники, статьи научных журналов, документацию библиотек) для полного понимания теоретических основ генеративных моделей. Выполняя проектные задания, студенты развивают навыки постановки задачи (создание изображения, генерация текста, синтез речи), подбора и предварительной обработки данных, выбора подходящей архитектуры и оценки результатов работы генеративной модели.

Важнейший компонент курса – самостоятельная проектная работа, в ходе которой студенты разрабатывают завершённое приложение на основе генеративных моделей. Эта работа позволяет закрепить навыки проектирования и реализации комплексных решений в области генеративного искусственного интеллекта. В результате студенты демонстрируют способности разработки полнофункциональных генеративных моделей для реальных задач.

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.

Кейсы ПАО «Сбербанк»

1. Генеративный ИИ для автоматического составления инвестиционных обзоров

Описание: Аналитики Сбера ежедневно составляют десятки аналитических и инвестиционных обзоров по рынкам, компаниям, макроэкономике. Задача – исследовать

применение LLM для генерации кратких сводок и аналитических отчетов на основе входных данных: биржевые котировки, макроэкономические показатели, рыночные события.

Цель: Разработать инструмент, способный по структурированным данным и краткому описанию формировать инвестиционный обзор в деловом стиле.

Ожидаемый результат: Модель, генерирующая аналитические тексты длиной 500–1000 слов с разделами «обзор событий», «рекомендации», «прогнозы», оформленные в формате банка.

2. Генерация сценариев фишинговых писем для обучения сотрудников

Описание: Банк проводит киберучения, включая рассылку тестовых фишинговых писем сотрудникам для повышения их устойчивости к социальным атакам. Проект предполагает использование генеративной модели для создания реалистичных фишинговых писем различных типов (поддельные счета, HR-запросы, ИТ-поддержка).

Цель: Создать генератор, способный на основе заданных параметров (тема, стиль, уровень угрозы) создавать тексты фишинга для тренировок.

Ожидаемый результат: Набор разнообразных примеров фишинга и оценка их эффективности по реакции сотрудников, а также классификация моделей угроз.

3. Генерация пользовательских сценариев работы в мобильном приложении

Описание: Банк хочет использовать генеративный ИИ для быстрой симуляции пользовательских сценариев – например, как клиент оформляет вклад, переводит средства, получает уведомление о риске мошенничества.

Цель: Разработать генератор пошаговых сценариев пользовательского поведения с вариативностью (молодой клиент, пенсионер, ИП).

Ожидаемый результат: Набор автоматически сгенерированных UX-сценариев, оформленных в виде сценариев для QA или UX-исследований, с логикой действий и типичными ошибками пользователя.

4. Генерация synthetic data для банковских моделей

Описание: Модели в Сбере требуют большого объема транзакционных и клиентских данных, которые нельзя использовать напрямую из-за требований ЦБ и ФЗ-152. Задача – разработать метод генерации синтетических банковских данных, максимально близких к реальным по распределениям и поведению.

Цель: Создать безопасный pipeline генерации данных (например, транзакций, профилей клиентов, шаблонов расходов) для обучения моделей.

Ожидаемый результат: Синтетический датасет и отчет о метриках приближенности к реальному (TSNE, K-L divergence и др.), с оценкой пригодности для обучения скоринговых или антифрод-моделей.

5. Улучшение персонализации обслуживания клиентов Сбербанка с помощью генеративных моделей

Описание: Сбербанк стремится значительно улучшить взаимодействие с клиентами, используя генеративные технологии для анализа поведения пользователей и построения персонализированных предложений и коммуникаций. Текущие модели недостаточно гибко адаптируются к изменениям предпочтений клиента, а также требуют больших вычислительных ресурсов для обработки большого объема разнородных данных.

Цель: Разработать решение на основе эксперт-ин-луп-подхода (Expert-in-the-loop), которое сможет учитывать интересы клиентов, предлагая индивидуальные рекомендации по продуктам банка, улучшив клиентский опыт и увеличив лояльность аудитории.

Ожидаемые результаты:

- Повышение уровня удовлетворенности клиентов услугами Сбербанка благодаря персональному подходу.
- Увеличение конверсии предложенных банковских продуктов и сервисов.
- Сокращение затрат на маркетинг и привлечение клиентов за счёт роста органической лояльности.

6. Автоматическое создание видеоконтента для продвижения банковских продуктов

Описание: Финансовое подразделение Сбербанка активно развивает цифровые каналы взаимодействия с пользователями. Необходимо создать автоматизированную систему, способную быстро создавать высококачественный рекламный контент (видео, баннеры, тексты) на основе заданных требований и характеристик продукта. Решение должно обеспечить высокую степень креативности и оперативность производства рекламных материалов.

Цель: Создать автоматический генератор контента, основанный на технологиях генеративных состязательных сетей (GANs) и мультимодальных трансформерах, который позволит оперативно формировать рекламные материалы высокого качества, снижая затраты на производство и повышая привлекательность бренда.

Ожидаемые результаты:

- Создание высокоэффективного и экономически выгодного инструмента для автоматизации маркетинга.
- Ускорение вывода на рынок новых финансовых продуктов.
- Рост узнаваемости брендов и повышение интереса клиентов к услугам Сбербанка.

7. Разработка виртуального ассистента для поддержки клиентов

Описание: Сбербанк испытывает растущую нагрузку на службу поддержки клиентов. Требуется внедрение инновационного решения, позволяющего автоматически отвечать на запросы клиентов, анализировать эмоции и поведение, предоставляя пользователям удобный интерфейс общения и минимизируя необходимость привлечения сотрудников службы поддержки.

Цель: Создание голосового ассистента, использующего генеративные нейронные сети и multimodal transformers для эффективной коммуникации с клиентом. Ассистент должен уметь поддерживать диалог, решать базовые задачи и предлагать полезные советы клиентам.

Ожидаемые результаты:

- Освобождение операторов кол-центра от рутинных запросов и повышение их производительности.
- Значительное улучшение качества обслуживания клиентов и снижение нагрузки на сервисные подразделения.
- Возможность эффективного расширения каналов общения с клиентами через голосовых помощников.

8. Генерация сценариев для обучения сотрудников

Описание: Сбербанк проводит обучение сотрудников по различным аспектам работы, включая обслуживание клиентов и работу с продуктами банка. Задача – разработать модель, которая будет генерировать сценарии для обучения сотрудников.

Цель: Создать модель, которая сможет генерировать сценарии, соответствующие различным ролям и задачам сотрудников.

Ожидаемый результат: Генеративная модель, способная создавать сценарии для обучения сотрудников, что повысит их квалификацию и эффективность работы.

9. Генерация рекомендаций для инвестиционных продуктов

Описание: Сбербанк предлагает широкий спектр инвестиционных продуктов, и важно предоставлять клиентам рекомендации, соответствующие их целям и риск-профилю. Задача – разработать модель, которая будет генерировать рекомендации для инвестиционных продуктов на основе данных о клиентах.

Цель: Создать модель, которая сможет анализировать данные о клиентах и генерировать рекомендации, соответствующие их инвестиционным целям и риск-профилю.

Ожидаемый результат: Генеративная модель, способная создавать рекомендации для инвестиционных продуктов, что повысит удовлетворенность клиентов и увеличит объем продаж.

10. Генерация контента для социальных сетей

Описание: Сбербанк активно использует социальные сети для взаимодействия с клиентами и продвижения своих продуктов. Задача – разработать модель, которая будет генерировать контент для социальных сетей на основе заданных параметров и целевой аудитории.

Цель: Создать модель, которая сможет генерировать тексты, изображения и видео для социальных сетей, соответствующие требованиям банка.

Ожидаемый результат: Генеративная модель, способная создавать контент для социальных сетей, что повысит вовлеченность аудитории и увеличит охват.

Кейсы от «АВАЛАБ»

1. Генеративный ИИ для создания проектной документации по ТЗ

Описание: В рамках проектирования объектов девелоперской компании архитекторы и инженеры тратят значительное время на подготовку текстовой проектной документации (обоснование решений, пояснительные записки, описания инженерных систем). Задача – исследовать возможность использования LLM для генерации черновиков проектной документации на основе исходных данных: этажность, материалы, климат, назначение, нормы.

Цель: Разработать прототип текстового генератора, который помогает специалистам быстрее формировать документацию в соответствии с шаблонами и нормативами.

Ожидаемый результат: Инструмент на основе LLM, создающий логически стройный и нормативно грамотный текст, поддающийся быстрой редакции инженером.

2. Генерация рекламного контента для жилых комплексов

Описание: «АВА ГРУПП» регулярно запускает маркетинговые кампании для жилых комплексов. Необходимо исследовать использование диффузионных моделей для генерации изображений (визуализации интерьеров, окрестностей, видов из окон) и LLM – для описаний квартир, преимуществ района, инфраструктуры.

Цель: Создать инструменты для быстрой генерации продающих материалов без привлечения дизайнеров и копирайтеров на первых этапах.

Ожидаемый результат: Набор сгенерированных карточек объектов с текстом, изображением и логикой «живого» рекламного сообщения.

3. Генерация документации и шаблонов договоров

Описание: Юридический департамент регулярно работает с договорами долевого участия, актами приёма-передачи и другими документами. Использование LLM может значительно сократить время на подготовку черновиков – достаточно ввести параметры сделки.

Цель: Создать систему, которая генерирует адаптированные тексты документов по вводным данным (тип объекта, этаж, площадь, ФИО, сроки и пр.).

Ожидаемый результат: Генератор документов в формате Word или PDF с автоматической подстановкой параметров и соблюдением юридического стиля.

4. Обратная генерация – ИИ-помощник для покупателей квартир

Описание: Будущие покупатели часто задают типовые вопросы о квартирах, планировках, ипотеке, акциях, сроках. Вместо call-центра предлагается реализовать LLM-бота, который обрабатывает текстовые и голосовые запросы, показывает планировки, ссылается на PDF-документы и может «объяснять» информацию простым языком.

Цель: Упростить коммуникацию с клиентами на этапе выбора квартиры и повысить качество первичного контакта.

Ожидаемый результат: Демо-бот, способный отвечать на вопросы о жилом комплексе, ориентируясь в его характеристиках и маркетинговых документах.

5. Генерация контента для маркетинговых кампаний

Описание: Девелоперская компания проводит маркетинговые кампании для продвижения своих проектов. Задача – разработать модель, которая будет генерировать контент для маркетинговых кампаний на основе заданных параметров и целевой аудитории.

Цель: Создать модель, которая сможет генерировать тексты, изображения и видео для маркетинговых кампаний, соответствующие требованиям компании.

Ожидаемый результат: Генеративная модель, способная создавать контент для маркетинговых кампаний, что повысит эффективность кампаний и привлечет больше клиентов.

6. Генерация сценариев для обучения сотрудников

Описание: Девелоперская компания проводит обучение сотрудников по различным аспектам работы, включая обслуживание клиентов и работу с продуктами компании. Задача – разработать модель, которая будет генерировать сценарии для обучения сотрудников.

Цель: Создать модель, которая сможет генерировать сценарии, соответствующие различным ролям и задачам сотрудников.

Ожидаемый результат: Генеративная модель, способная создавать сценарии для обучения сотрудников, что повысит их квалификацию и эффективность работы.

7. Генерация контента для социальных сетей

Описание: Девелоперская компания активно использует социальные сети для взаимодействия с клиентами и продвижения своих продуктов. Задача – разработать модель, которая будет генерировать контент для социальных сетей на основе заданных параметров и целевой аудитории.

Цель: Создать модель, которая сможет генерировать тексты, изображения и видео для социальных сетей, соответствующие требованиям компании.

Ожидаемый результат: Генеративная модель, способная создавать контент для социальных сетей, что повысит вовлеченность аудитории и увеличит охват.

8. Построение индивидуальных жилищных пространств с применением генеративного дизайна

Описание: Девелоперская компания сталкивается с задачей создания уникальных жилых помещений, отражающих предпочтения конкретного покупателя недвижимости. Рынок жилья становится всё более требовательным к индивидуальному дизайну квартир и домов. Однако традиционный подход проектирования занимает много времени и ограничивает возможности разработчиков по созданию уникального продукта.

Цель: Разработать инструмент на основе генеративных состязательных сетей (GANs) и диффузионных моделей, позволяющий автоматически проектировать уникальные жилые помещения с учётом пожеланий заказчика, учитывая при этом нормативные требования и технические ограничения строительства.

Ожидаемые результаты:

- Повышение привлекательности жилого фонда компании среди покупателей элитной недвижимости.
- Снижение временных издержек на проектирование уникальных объектов.
- Формирование имиджа компании как лидера инновационных подходов в области жилой застройки.

9. Интеграция мультимодальных моделей для планирования городских территорий

Описание: Компания занимается развитием районов и застройкой целых кварталов, что предполагает принятие множества взаимосвязанных решений относительно инфраструктуры, транспортной доступности, экологии и благоустройства территории. Чтобы избежать ошибок и минимизировать риски, необходим мощный аналитический инструмент, способный совмещать разнотипные данные и выдавать оптимальное комплексное решение.

Цель: Создать интерактивную среду планирования городской среды на основе мультиспектральных трансформеров, обрабатывающих текстовую, визуальную и звуковую информацию, такую как планы улиц, карты рельефа местности, климатические условия и даже записи социальных настроений жителей района.

Ожидаемые результаты:

- Эффективное управление ресурсами при строительстве новых районов и зданий.
- Повышение удобства проживания и привлекательного образа городов, формируемых компаниями.
- Минимизация рисков неудачных строительных проектов за счёт комплексного подхода к принятию решений.

10. Инновационная система безопасности объектов недвижимости

Описание: Современные объекты недвижимости нуждаются в усилении мер безопасности. Традиционные охранные системы часто оказываются неэффективными против современных угроз, особенно киберугроз и преступлений, связанных с мошенничеством и несанкционированным проникновением.

Цель: Разработайте интеллектуальную систему безопасности на основе генеративных состязательных сетей (GANs) и автокодировщиков, которая способна обнаруживать аномалии и угрозы в режиме реального времени, предупреждая злоумышленнические действия ещё до их совершения.

Ожидаемые результаты:

- Повышение уровня защищённости объектов недвижимости компании.
- Укрепление репутации компании как надежного застройщика, предлагающего жильё премиум-класса с повышенной защитой.
- Привлечение новых категорий потребителей, ценящих безопасность и конфиденциальность.

7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю)

7.1 Перечень информационно-коммуникационных технологий

1. Облачные платформы и сервисы

cloud.ru, YandexCloud, AWS/GCP/Azure – облачные вычисления

Google Colab – облачная среда для выполнения кода на Python с GPU/TPU

Kaggle – платформа для работы с датасетами и соревнований по ML

2. Системы управления версиями и коллаборации

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

Notion/Trello – организация проектной деятельности

3. Система управления обучением

Moodle – сдача работ

7.2 Перечень лицензионного и свободно распространяемого программного обеспечения

1. Лицензионное ПО

VSCoде – IDE для Python

LibreOffice – оформление отчетов (свободнораспространяемое)

2. Свободное ПО (Open Source)

JupyterHub – для работы с jupyter notebook

Библиотеки:

numpy – работа с матрицами

pandas – работа с датасетами (загрузка из файлов, предобработка)

scikit-learn – разработка простых нейросетей

PyTorch (>= 1.10) - основная платформа для реализации глубоких нейронных сетей, позволяющая строить как классические, так и новые продвинутые архитектуры (GAN, VAE, Diffusion models).

OpenCV/Torchvision - для обработки изображений, загрузки датасетов и предварительной обработки.

SpaCy/NLTK/Gensim - для предварительной обработки текстовых данных (токенизация, векторизация и другие операции над текстом).

Transformers (HuggingFace) - для работы с текстовыми данными, основанными на архитектуре трансформеров (BERT, GPT и т.д.). Используется для создания мультимодальных моделей и кросс-модальных приложений.

Clip - Официальная реализация модели CLIP и её аналогов (например, ruCLIP) для связывания изображений и текста.

Diffusers (HuggingFace) - Библиотека для работы с диффузионными моделями (например, Stable Diffusion).

DeepProbLog/Neurosymbolic AI Toolkit – нейро-символические фреймворки

Ortuna – открытая библиотека на языке Python, предназначенная для эффективного поиска оптимальных гиперпараметров моделей машинного обучения и глубокого обучения.

GluonCV/GluonNLP - дополнительные утилиты для упрощенной работы с CV и NLP-датасетами и моделями.

Librosa/Madmom/Audiomentations - Специальные библиотеки для работы с аудиоданными, применяемые для обработки сигналов, нормализации и выделения характеристик (MFCC, STFT и т.д.).

Tacotron/Wav2Vec/ESPnet - Популярные open-source библиотеки для задач преобразования текста в речь (Text-to-Speech, TTS) и обратно (Automatic Speech Recognition, ASR).

Whisper - Современная система транскрипции речи от OpenAI, работающая на архитектуре трансформеров.

SoundFile/audioio - Низкоуровневые библиотеки для чтения и записи звуковых файлов.

Инструменты для визуализации:

Matplotlib/Seaborn – графики и анализ данных

СУБД:

SQLite/PostgreSQL – хранение структурированных данных

FAISS/Annoy – векторный поиск

8. Материально-техническое обеспечение по дисциплине (модулю)

Виртуальные машины 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 |
| 2 | Виртуальная машина с GPU | Аренда публичного IP | 1 | | Шт |
| | | 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 | | |
| 3 | K8S | Аренда публичного IP | 1 | | Шт |
| | | 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 |
| 4 | ML Inference Instance Type GPU | Аренда публичного IP | 1 | | Шт |
| | | 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.

| № | Вид работ | Наименование учебной аудитории, ее оснащенность оборудованием и техническими средствами обучения |
|----|----------------------|---|
| 1. | Лекционные занятия | Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения |
| 2. | Лабораторные занятия | Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, проектором, программным обеспечением |

| | | |
|----|--|--|
| 3. | Практические занятия | Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения |
| 4. | Групповые (индивидуальные) консультации | Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением |
| 5. | Текущий контроль, промежуточная аттестация | Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением |
| 6. | Самостоятельная работа | Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета. |

Примечание: Конкретизация аудиторий и их оснащение определяется ОПОП.