

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

УТВЕРЖДАЮ

Проректор по учебной работе,
качеству образования – первый
проректор

Хагуров Т.А.

подпись

«02» июня 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.16 Нейросетевые технологии

Направление подготовки 02.03.03 Математическое обеспечение и
администрирование информационных систем

Направленность (профиль) Искусственный интеллект и аналитика данных

Форма обучения очная

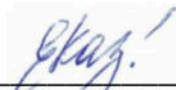
Квалификация бакалавр

Краснодар 2026

Рабочая программа дисциплины «Нейросетевые технологии» 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):

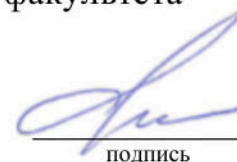
Е.В. Казаковцева, доцент кафедры анализа данных и
искусственного интеллекта, к. ф.-м. н.


подпись

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


подпись

Утверждена на заседании учебно-методической комиссии факультета
компьютерных технологий и прикладной математики
протокол № 03 «15» мая 2026 г.
Председатель УМК факультета Добровольская Н.Ю.


подпись

Рецензенты:

Мостовой Евгений Викторович, генеральный директор ООО «Портал-Юг»,
e-mail: mostovoy@portal-yug.ru

Луценко Евгений Вениаминович, доктор экономических наук, кандидат
технических наук, профессор кафедры компьютерных технологий и систем
Федерального государственного 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

Развитие у студентов компетенций в области анализа и интерпретации механизмов работы ключевых видов нейронных сетей, подготовки и обработки данных для эффективного нейросетевого моделирования, освоения методов генетической оптимизации, адаптации моделей к новым данным, увеличения разнообразия выборок посредством аугментации, объединения результатов моделей с использованием ансамблевых подходов, расчета и анализа показателей качества моделей, необходимых специалистам в сфере аналитики данных, разработки AI-решений, управления ML-инфраструктурой и руководстве проектами в области искусственного интеллекта.

1.2 Задачи дисциплины

1. Подготовка данных для нейросетевого моделирования, аугментация и ансамблирование данных
2. Анализ архитектур нейронных сетей (автокодировщики, графовые нейросети, неглубокие нейросети для работы с текстом и изображениями и т.д.).
3. Изучение методов адаптивного обучения и генетических алгоритмов.
4. Практическое освоение инструментов реализации нейронных сетей на фреймворках tensorflow и keras, в библиотеке Spektral.

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Нейросетевые технологии» относится к части, формируемой участниками образовательных отношений Блока 1 «Дисциплины (модули)» учебного плана. Данная дисциплина в качестве прerreквизитов требует освоения следующих дисциплин: «Математические модели нейронных сетей» (основные алгоритмы обучения, функции активации, оптимизаторы, начальные знания об архитектурах SOM и RBF, интерпретируемости моделей, работа в jupyter notebook и google colab), «Технологии компьютерного зрения» (начальные сведения о работе с изображениями, фильтрации и задачах в CV, работа с gitlab), «Математический анализ» (основы дифференциального исчисления, интегральное исчисление и ряды, теория функций нескольких переменных), «Обработка данных на Python» (основы Python, работа с данными для подготовки обучающей выборки на Python), «Курс теории вероятностей» (распределения, матожидание, дисперсия и др. важные понятия), «Алгебра и геометрия» (операции с векторами и матрицами, системы линейных уравнений, проекции, ортогональность, преобразования координат). Постреквизитами выступают дисциплины, где продолжается изучение нейросетевых технологий и их применение на практике: «Глубокое обучение» (рассмотрение более сложных архитектур для работы с изображениями и текстом, освоение новых архитектур и техник глубокого обучения, реализация через pytorch, tensorflow, keras), «MLOps&DevOps» (внедрение нейросетей), «Методы искусственного интеллекта в задачах классификации» (применение на практике изученных архитектур в задачах классификации), «ИИ Финтех» (практическая реализация рассмотренных архитектур в финансовом секторе), «Искусственный интеллект в оценке рисков и разработке страховых продуктов» (внедрение нейросетей в процессы разработки страховых продуктов), «Генеративный искусственный интеллект» (более сложные архитектуры для работы с текстами и изображениями, работа с несколькими модальностями), «Технологии обработки языка, звуковых данных, включая распознавание и синтез речи» (применение на практике изученных принципов обучения, освоение новых архитектур для работы с текстами и аудио). Смежными дисциплинами, изучаемыми параллельно или логически близко,

являются «Управление жизненным циклом ИИ-продукта» (применение изученных архитектур), «Машинное обучение» (подготовка данных, методы кластеризации).

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
MF-3 Способен применять современные методы оптимизации для обучения моделей машинного обучения, настройки гиперпараметров и решения задач искусственного интеллекта.	MF-3.2 Применяет методы оптимизации для настройки гиперпараметров моделей машинного обучения, включая использование методов поиска (grid search, random search) и байесовской оптимизации	(С) Умеет настраивать гиперпараметры с использованием более сложных методов, таких как байесовская оптимизация, для улучшения производительности моделей и минимизации времени обучения.
ML-4 Способен применять методы обучения без учителя для анализа структуры данных и выявления скрытых закономерностей	ML-4.1 Применяет алгоритмы кластеризации и понижения размерности для решения практических задач	(С) Владеет инструментами очистки данных и предварительной подготовки данных методами понижения размерности и визуализации для анализа данных.
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	(С) Задает скорость обучения в зависимости от задачи и набора данных. Выбирает функцию потерь в зависимости от задачи и набора данных. Способен

	обратного распространения ошибки (backpropagation), для эффективного обучения моделей	применять регуляризацию и прореживание. Выбирает размер пакета для стохастического градиентного спуска; понимает принцип градиентного спуска
	DL-1.2 Способен проектировать и реализовывать неглубокие нейронные сети (перцептроны, MLP), выбирать appropriate функции активации и регуляризации для решения задач классификации и регрессии	(C) Способен разрабатывать и/или применять самоорганизующиеся карты Кохонена. Способен разрабатывать RBF-сети (сети регуляризации, обобщенные RBF-сети)
	DL-1.5 Разрабатывает и оптимизирует специализированные архитектуры для работы с последовательностями, учитывая их уникальные свойства	(C) Понимает принцип работы блока единичной задержки в рекуррентных нейронных сетях; применяет блоки RNN, GRU, LSTM для решения задач
	DL-1.6 Разрабатывает, адаптирует и внедряет генеративные нейронные сети для решения практических задач, включая создание новых архитектур, оптимизацию обучения и промышленное развертывание моделей	(C) Модифицирует архитектуры под специфические требования; разрабатывает комплексные пайплайны обучения; применяет продвинутое техники - перенос обучения, генерация по нескольким примерам
	DL-1.9 Разрабатывает, адаптирует и внедряет трансформерные архитектуры для решения задач обработки последовательностей, включая создание новых моделей, оптимизацию обучения и промышленное развертывание	(C) Понимает и применяет специфические архитектурные варианты для задач (например, вариант "энкодер-декодер" для перевода/суммаризации, вариант "только энкодер" для генерации); использует для работы с длинными последовательностями эффективные схемы внимания (Lformer, Performer, Longformer, Sparse Transformers).

		Проводит детальный анализ ошибок модели, понимает типичные проблемы (смещение в данных/модели, переобучение) и применяет методы борьбы с ними (регуляризация, аугментация данных)
ФС-5 Способен проводить фронтальные исследования в области безопасности, доверия и объяснимости	ФС-5.2 Обеспечивает объяснения причин принятия тех или иных решений в результатах работы искусственного интеллекта	(С) Создает методы post-factum объяснения (интерпретации) непрозрачных методов искусственного интеллекта на основе нейронных сетей

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 4 зач. ед. (144 часа), их распределение по видам работ представлено в таблице

Вид учебной работы	Всего часов	Семестры (часы)					
		5					
Контактная работа, в том числе:	52.3	52.3					
Аудиторные занятия (всего):	50	50					
Занятия лекционного типа	16	16					
Лабораторные занятия	34	34					
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:	56	56					
Курсовая работа	-	-					
Проработка учебного (теоретического) материала	26	26					
Выполнение индивидуальных заданий (подготовка сообщений, презентаций)	-	-					
Реферат	-	-					
Подготовка к текущему контролю	30	30					
Контроль:	35,7	35,7					
Подготовка к экзамену	35,7	35,7					
Общая трудоемкость	час.	144	144				
	в том числе контактная работа	52.3	52.3				
	зач. Ед	4	4				

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.
Разделы (темы) дисциплины, изучаемые в 5 семестре

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
1.	Подготовка данных для нейросетевого моделирования	13	3		4	6
2.	Архитектуры нейронных сетей. Метрики	35	6		10	19
3.	Адаптивное обучение нейросетей	15	1		6	8
4.	Инструменты разработки нейросетей на TensorFlow и Keras	16	2		4	10
5.	Применение нейросетевых технологий в финансах	14	3		4	7
	Интерпретация результатов и повышение доверительности AI-моделей.	13	1		6	6
ИТОГО по разделам дисциплины		106	16		34	56
Контроль самостоятельной работы (КСР)		2				
Промежуточная аттестация (ИКР)		0,3				
Подготовка к экзамену		35,7				
Общая трудоемкость по дисциплине		144				

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.	Подготовка данных для нейросетевого моделирования	Основы предобработки данных: очистка данных, заполнение пропусков, нормализация и масштабирование признаков. Аугментация данных: алгоритмы аугментации изображений и текста, инструменты расширения выборки. Методы ансамбля данных: бэггинг, бутстрэп, bootstrapping, stacking, blending.	Т
2.	Архитектуры нейронных сетей. Метрики	Автокодировщики: типы автокодировщиков, применение в сжатии данных и извлечении признаков. Графовые нейросети: архитектура и использование в задаче классификации графа.	Т

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
		Неглубокие нейросети для работы с текстом и изображениями: архитектуры, особенности, оптимизация. Введение в генеративные нейросети и техники переноса обучения. Генетические алгоритмы Метрики оценки качества нейронных сетей в задачах регрессии, классификации, кластеризации и понижения размерности	
3.	Адаптивное обучение нейросетей	Автоматическое дифференцирование. Функции потерь в нейронных сетях, модернизированные алгоритмы оптимизации Современные методы адаптации нейросетей: MAML (Model-Agnostic Meta-Learning).	Т
4.	Инструменты разработки нейросетей на TensorFlow и Keras	Особенности работы с библиотеками TensorFlow и Keras: структура API, настройка гиперпараметров, Grid Search, Random Search. Самообучающиеся структуры: самоорганизующиеся карты Кохонена, Expanding Neuron Gas, Radial Basis Function (RBF)-сети.	Т
5.	Применение нейросетевых технологий в финансах	Прогнозирование финансовых временных рядов: введение в финансовые временные ряды, классические модели и перспективы применения нейросетей. Оценка рисков и управление рисками в финансовой сфере с применением нейросетевых ансамблей.	Т
6.	Интерпретация результатов и повышение доверительности AI-моделей	Основные техники интерпретируемости нейросетей: SHAP, LIME, Partial Dependence Plots. Объяснимый ИИ и механизмы увеличения прозрачности нейросетевых моделей.	Т

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.	Подготовка данных для нейросетевого моделирования	Лабораторная работа № 1. Предобработка данных Лабораторная работа №2. Аугментация данных Лабораторная работа №3. Ансамблирование данных	ЛР
2.	Архитектуры нейронных сетей. Метрики	Лабораторная работа №4. Простое построение нейросети Лабораторная работа №5. Автоэнкодеры Лабораторная работа №6. Работа с графовыми нейросетями Лабораторная работа №7. Легковесные нейросети для текста и изображений	ЛР
3.	Адаптивное обучение нейросетей	Лабораторная работа №8. Выбор гиперпараметров и оптимизационные стратегии	ЛР
4.	Инструменты разработки нейросетей на TensorFlow и Keras	Лабораторная работа №9. Гиперпараметрическая оптимизация Лабораторная работа №10. Самоорганизующиеся сети	ЛР
5.	Применение нейросетевых технологий в финансах	Лабораторная работа №11. Прогнозирование временных рядов Лабораторная работа №12. Управление кредитными рисками	ЛР
6.	Интерпретация результатов и повышение доверительности AI-моделей	Лабораторная работа №13. Интерпретация предсказаний моделей	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
5	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 «Нейросетевые технологии».

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	Подготовка данных для нейросетевого моделирования	ML-4.1 (Средний)	Тестирование Лабораторные работы №1-3	Вопросы к экзамену 3, 4, 21, 26
2	Архитектуры нейронных сетей. Метрики	DL-1.1 (Средний); DL-1.2 (Средний); DL-1.5 (Средний); DL-1.6 (Средний); DL-1.9 (Средний)	Тестирование, Лабораторные работы №4-7	Вопросы к экзамену 1-3, 9, 11, 27-30
3	Адаптивное обучение нейросетей	MF-3.2 (Средний)	Тестирование Лабораторная работа №8	Вопросы к экзамену 5,6
4	Инструменты разработки нейросетей на TensorFlow и Keras	MF-3.2 (Средний); ML-4.1 (Средний); DL-1.1 (Средний); DL-1.2 (Средний); DL-1.5 (Средний); DL-1.6 (Средний); DL-1.9 (Средний)	Тестирование, Лабораторные работы №9-10	Вопросы к экзамену 7,8, 12-17, 22-24
5	Применение нейросетевых технологий в финансах	DL-1.1 (Средний), DL-1.5 (Средний), DL-1.6 (Средний)	Тестирование, Лабораторные работы №11-12	Вопросы к экзамену 25
6	Интерпретация результатов и повышение доверительности AI-моделей	FC-5.2 (Средний)	Тестирование, Лабораторная работа №13	Вопросы к экзамену 18-20

Показатели, критерии и шкала оценки сформированных компетенций

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

MF-3 *Способен применять современные методы оптимизации для обучения моделей машинного обучения, настройки гиперпараметров и решения задач искусственного интеллекта.*

- Умеет настраивать гиперпараметры с использованием более сложных методов, таких как байесовская оптимизация, для улучшения производительности моделей и минимизации времени обучения.
- ML-4** ***Способен применять методы обучения без учителя для анализа структуры данных и выявления скрытых закономерностей***
Владеет инструментами очистки данных и предварительной подготовки данных методами понижения размерности и визуализации для анализа данных.
- DL-1** ***Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей***
Задаёт 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. Способен разрабатывать и/или применять самоорганизующиеся карты Кохонена. Способен разрабатывать RBF-сети (сети регуляризации, обобщенные RBF-сети).
Понимает принцип работы блока единичной задержки в рекуррентных нейронных сетях; применяет блоки RNN, GRU, LSTM для решения задач. Модифицирует архитектуры под специфические требования; разрабатывает комплексные пайплайны обучения; применяет продвинутое техники - перенос обучения, генерация по нескольким примерам.
Понимает и применяет специфические архитектурные варианты для задач (например, вариант "энкодер-декодер" для перевода/суммаризации, вариант "только энкодер" для генерации); использует для работы с длинными последовательностями эффективные схемы внимания (Linformer, Performer, Longformer, Sparse Transformers). Проводит детальный анализ ошибок модели, понимает типичные проблемы (смещение в данных/модели, переобучение) и применяет методы борьбы с ними (регуляризация, аугментация данных).
- FC-5** ***Способен проводить фронтальные исследования в области безопасности, доверия и объяснимости***
Создаёт методы post-factum объяснения (интерпретации) непрозрачных методов искусственного интеллекта на основе нейронных сетей.
Владеет методами верификации авторства результатов работы нейросетей.

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

- MF-3** ***Способен применять современные методы оптимизации для обучения моделей машинного обучения, настройки гиперпараметров и решения задач искусственного интеллекта.***
Знает и использует стандартные методы поиска гиперпараметров, такие как grid search и random search, для настройки моделей машинного обучения в стандартных задачах.
Умеет настраивать гиперпараметры с использованием более сложных методов, таких как байесовская оптимизация, для улучшения производительности моделей и минимизации времени обучения.
- ML-4** ***Способен применять методы обучения без учителя для анализа структуры данных и выявления скрытых закономерностей***

Применяет стандартные библиотеки для обучения без учителя (ScikitLearn), понимает базовые модели для кластерного анализа (иерархическая кластеризация - k-средних, DBSCAN, иерархическая кластеризация), методы понижения размерности (PCA, t-SNE)

Владеет инструментами очистки данных и предварительной подготовки данных методами понижения размерности и визуализации для анализа данных.

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

Задаёт 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. Способен разрабатывать и/или применять самоорганизующиеся карты Кохонена. Способен разрабатывать RBF-сети (сети регуляризации, обобщённые RBF-сети)

Понимает принцип работы блока единичной задержки в рекуррентных нейронных сетях; применяет блоки RNN, GRU, LSTM для решения задач

Модифицирует архитектуры под специфические требования; разрабатывает комплексные пайплайны обучения; применяет продвинутое техники - перенос обучения, генерация по нескольким примерам

Понимает и применяет специфические архитектурные варианты для задач (например, вариант "энкодер-декодер" для перевода/суммаризации, вариант "только энкодер" для генерации); использует для работы с длинными последовательностями эффективные схемы внимания (Linformer, Performer, Longformer, Sparse Transformers). Проводит детальный анализ ошибок модели, понимает типичные проблемы (смещение в данных/модели, переобучение) и применяет методы борьбы с ними (регуляризация, аугментация данных)

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

Способен реализовывать многослойный персептрон

Выбирает значения гиперпараметров нейронов рекуррентных сетей; применяет готовые архитектуры рекуррентных нейронных сетей LSTM, GRU

Умеет применять готовые генеративные модели (GAN, VAE, Diffusion) через API и библиотеки. Проводит базовое дообучение предобученных моделей. Использует стандартные метрики оценки качества генерации

Понимает принципы работы трансформеров (самовнимание, позиционное кодирование). Умеет применять готовые модели (BERT, GPT, ViT) через Hugging Face. Настраивает базовые параметры обучения (скорость обучения, размер пакета)

FC-5 ***Способен проводить фронтальные исследования в области безопасности, доверия и объяснимости***

Умеет применять подходы для обеспечения объяснимости, повышения доверия работы искусственного интеллекта

Владеет стандартными инструментами объяснения в машинном обучении для интерпретации предсказаний моделей (SHAP, LIME и т.п.)

Создаёт методы post-factum объяснения (интерпретации) непрозрачных методов искусственного интеллекта на основе нейронных сетей

Владеет методами верификации авторства результатов работы нейросетей

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

MF-3 *Способен применять современные методы оптимизации для обучения моделей машинного обучения, настройки гиперпараметров и решения задач искусственного интеллекта.*

Разрабатывает и применяет собственные методы оптимизации гиперпараметров для сложных задач с высокой размерностью, использует адаптивные подходы и интегрирует их с моделями машинного обучения для повышения точности и скорости работы

ML-4 *Способен применять методы обучения без учителя для анализа структуры данных и выявления скрытых закономерностей*

Выбирает и настраивает алгоритмы кластеризации (DBSCAN, сдвиг среднего, гауссовские смеси) и методы понижения размерности (UMAP, автоэнкодеры) в зависимости от специфики задачи. Интерпретирует полученные результаты и применяет их для обоснованных выводов

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

Применяет оптимизаторы к функции потерь во избежание проблемных ситуаций на ландшафте функции потерь (например, овраги, седловые точки и т.п.). Визуализирует ландшафт функции потерь. Внедряет пакетную нормализацию в архитектуру нейронной сети. Применяет для обучения нейронных сетей методы оптимизации второго порядка (L-BFGS, Левенберга-Марквардта, квазиньютоновские методы, методы Ньютона). Разрабатывает байесовские нейронные сети и применяет вариационный вывод для их обучения

Владеет способами борьбы с перекрутом в сетях SOM; знает принципы построения разделяющих гиперповерхностей; способен разрабатывать ограниченные машины Больцмана

Понимает принципы функционирования и обучения фильтров и ячейки памяти в GRU и LSTM блоках; понимает принцип обучения с помощью обратного распространения по времени

Понимает принципы работы генеративных сетей, адаптивно применяет архитектуру VAE+GAN; понимает как разрабатывать капсульные сети

Понимает принцип работы многоголовочного внимания

FC-5 *Способен проводить фронтальные исследования в области безопасности, доверия и объяснимости*

Создает новые методы прозрачного искусственного интеллекта на основе формальных систем, логики и баз знаний

Разрабатывает общие требования и протоколы тестирования систем ИИ на объяснимость

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 является целью процесса нормализации данных?

А. Удаление выбросов и аномалий

- B. Масштабирование всех признаков к одному диапазону
 - C. Повышение количества наблюдений в выборке
 - D. Изменение порядка следования элементов в выборке
-

Вопрос 2: Какой метод предназначен для автоматического добавления разнообразия в данные путём модификации существующих объектов?

- A. Over-sampling
 - B. Augmentation
 - C. Under-sampling
 - D. Clustering
-

Вопрос 3: Какие данные называются "пропущенными"?

- A. Данные, относящиеся к другим категориям
 - B. Некорректные данные
 - C. Незначительные для модели данные
 - D. Отсутствие информации по некоторым признакам в некоторых объектах
-

Вопрос 4: Для чего применяется метод bootstrap sampling?

- A. Для уменьшения размеров тренировочной выборки
 - B. Для формирования репрезентативных выборок путём случайного отбора с возвратом
 - C. Для фильтрации шумов из данных
 - D. Для удаления дубликатов из набора данных
-

Вопрос 5: Почему важно обрабатывать пропущенные значения в данных?

- A. Пропущенные значения замедляют вычисления
 - B. Модели часто плохо работают с отсутствующими значениями
 - C. Это улучшает читаемость данных
 - D. Без обработки повышается вероятность ложноположительных ошибок
-

Вопрос 6: Назначение техники аугментации данных в процессе обучения нейросети:

- A. Сделать модель менее чувствительной к изменениям среды
 - B. Снизить уровень шума в сигнале
 - C. Искусственно увеличить объем данных
 - D. Все перечисленное верно
-

Вопрос 7: Как называется процедура объединения множества небольших выборок в одну большую?

- A. Ensembling
 - B. Bootstrapping
 - C. Cross-validation
 - D. Aggregating
-

Вопрос 8: Основной недостаток подхода аугментации данных — это:

- A. Сложность вычислений
 - B. Потеря точности модели
 - C. Чрезмерное увеличение объемов данных
 - D. Возможное появление избыточных повторений данных
-

Вопрос 9: При выборе подходящего способа нормализации важно учитывать:

- A. Тип используемых данных
- B. Размер имеющегося набора данных
- C. Характеристики распределенности признаков

D. Всё вышеперечисленное

Вопрос 10: Автоматизированная обработка пропущенных значений возможна при помощи:

- A. Только среднего арифметического значения
- B. Замены на нулевое значение
- C. Среднего арифметического, медианного значения или режима (самого частого значения)
- D. Исключительно замены на произвольное число

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:
ML-4.1; MF-3.2; DL-1.1; DL-1.2; DL-1.5; DL-1.6; DL-1.9; FC-5.2

Практические кейсы по тематике лабораторных работ

Лабораторная работа №1: Предобработка данных

Кейс: Система мониторинга экологического состояния водоема

Постановка задачи: Вам поручено создать систему автоматизированного контроля загрязнений водоемов города N. Необходимо подготовить датасет химического состава воды (концентрация солей, pH, содержание тяжелых металлов и др.). Ваша задача — провести полную предобработку данных для последующего анализа загрязнения вод с помощью нейросети.

Требования:

- Загрузите датасет водных проб и произведите обработку данных (нормализацию, удаление пропусков, кодировку категорий).
- Проанализируйте корреляционную матрицу признаков и удалите сильно коррелированные признаки.
- Сделайте выводы о влиянии каждого признака на загрязненность водоёма.

Лабораторная работа №2: Аугментация данных

Кейс: Диагностика дефектов деталей на производстве

Постановка задачи: Необходимо автоматизировать диагностику повреждений деталей оборудования предприятия. Исходный датасет фотографий поврежденных изделий недостаточен для качественного обучения нейросети. Требуется расширить его путем аугментации изображений.

Требования:

- Примените технику аугментации данных (вращение, изменение контрастности, отражение, обрезка краев).
- Продемонстрируйте влияние аугментации на точность классификатора дефектов.

Лабораторная работа №3: Ансамблирование данных

Кейс: Автоматизированная система распознавания автомобильных марок

Постановка задачи: Вы создаёте приложение для быстрой идентификации марки автомобиля по фотографии. Для повышения точности диагностики нужно объединить несколько обучающих выборок в единый ансамбль данных.

Требования:

- Используйте bootstrapping для разделения общего датасета на подмножества.
- Создайте отдельные модели для каждого подмножества и объедините их прогнозы в единую модель-классификатор.
- Проверьте точность получившейся системы и сравните её с точностью обычной нейросети.

Лабораторная работа №4: Простое построение нейросети

Кейс: Определение категории заболевания по медицинским снимкам

Постановка задачи: Ваш отдел занимается разработкой программного комплекса для медицинской диагностики заболеваний легких по рентгенограммам. Нужно создать простую нейросеть для классификации снимков пациентов на две категории: здоров или болен пневмонией.

Требования:

- Постройте трёхслойную нейросеть с плотными слоями (Dense) и Softmax активацией на выходном слое.
- Произведите тестирование модели на специально выделенном контрольном наборе данных.
- Проведите оценку метрик

Лабораторная работа №5: Автоэнкодеры

Кейс: Реконструкция повреждённых документов

Постановка задачи: В архиве хранится большое количество исторических документов, многие из которых повреждены временем. Требуется восстановить утраченную информацию с помощью автоэнкодера.

Требования:

- Разработайте простое представление документа и обучите автоэнкодер восстанавливать оригинальные версии документов по искажённым копиям.
- Оцените качество восстановления.

Лабораторная работа №6: Работа с графовыми нейросетями

Кейс: Распознавание групп мошенников в социальной сети

Постановка задачи: Социальная сеть нуждается в инструменте для обнаружения преступных сообществ среди пользователей. Ваша задача — реализовать графовую нейросеть, способную анализировать связи пользователей и выявлять подозрительные группы.

Требования:

- Примените GCN для классификации пользователей на "добросовестных" и "сомнительных".
- Выявите сообщества, обладающие признаками мошенничества.

Лабораторная работа №7: Легковесные нейросети для текста и изображений

Кейс: Детектор поддельных фотографий

Постановка задачи: Нужно создать инструмент для быстрого выявления подделок цифровых изображений. Датасет содержит настоящие и фальшивые снимки знаменитостей.

Требования:

- Обучите лёгкую свёрточную нейросеть (LeNet) для детекции подлинности изображения.
- Определите, насколько точно модель способна отличать реальные фотографии от поддельных.

Лабораторная работа №8: Выбор гиперпараметров и оптимизационные стратегии

Кейс: Повышение точности рекомендаций продуктов магазина

Постановка задачи: Интернет-магазин хочет повысить эффективность своей рекомендательной системы. Вам предстоит оптимизировать гиперпараметры существующей нейросети, использующейся для персонализации предложений покупателям.

Требования:

Найдите оптимальное сочетание гиперпараметров (скорость обучения, размер партии, число эпох) с помощью экспериментов с разными конфигурациями. Представьте отчёт о влиянии выбранных параметров на итоговую точность рекомендаций.

Лабораторная работа №9: Гиперпараметрическая оптимизация

Кейс: Формирование индивидуальных маршрутов такси в городе

Постановка задачи: Такси-сервис хочет предложить своим пассажирам индивидуальные маршруты поездок. Необходимо настроить гиперпараметры нейросети, которая должна рассчитывать оптимальные пути перемещения водителей в течение рабочего дня.

Требования:

- Сравните методы Grid Search и Random Search для выбора лучшего сочетания гиперпараметров.
- Предоставьте рекомендации по выбору наилучшей комбинации гиперпараметров.

Лабораторная работа №10: Самоорганизующиеся сети

Кейс: Распознавание эмоций по голосовым сообщениям

Постановка задачи: Финансовая компания хочет внедрить сервис анализа эмоционального тона клиентов по телефонным разговорам. Вам необходимо построить SOFM для сегментации записей разговоров по эмоциональному окрасу голоса.

Требования:

- Реализуйте Self-Organizing Map для кластеризации голосов по эмоциям (радость, гнев, спокойствие).
- Постройте RBF-сеть для дальнейшего точного распознавания эмоций.

Лабораторная работа №11: Прогнозирование временных рядов

Кейс: Прогноз спроса на электроэнергию в регионе

Постановка задачи: Энергетическая компания ищет способ прогнозировать потребление электроэнергии жителями региона на ближайшие дни. Вам нужно создать модель для прогнозирования потребления энергии на основании предыдущих данных.

Требования:

- Обучите нейросеть на временном ряду энергопотребления и сделайте прогноз на следующий период.
- Оцените точность прогноза и дайте рекомендации по улучшению модели.

Лабораторная работа №12: Управление кредитными рисками

Кейс: Оценка платежеспособности клиента

Постановка задачи: Банк внедряет новую систему оценки кредитных рисков. Ваше задание — создать нейросеть для оценки вероятности возврата кредита клиентом. Используется датасет кредитной истории физических лиц.

Требования:

- Примените ансамбль нейросетей для классификации потенциальных заемщиков на надёжных и ненадёжных.
- Просчитайте коэффициент вероятности невозврата кредита каждым классом клиентов.

Лабораторная работа №13: Интерпретация предсказаний моделей

Кейс: Поиск причины отказов медицинского оборудования

Постановка задачи: Медицинская организация стремится определить причину выхода из строя диагностического оборудования. Необходимо проанализировать большой массив эксплуатационной статистики с помощью нейросети и представить объяснение вывода нейросети.

Требования:

- Используйте метод SHAP для анализа вклада признаков в конечный диагноз отказа оборудования.
- Интерпретируйте полученные результаты и представьте рекомендацию по устранению неисправности.

Пример лабораторной работы *Лабораторная работа №5. Автоэнкодеры*

Цель:

Освоить принципы работы автоэнкодеров, научиться проектировать и обучать глубокие и простые нейросети для сжатия и восстановления изображений. Изучить влияние архитектуры и числа слоев на качество реконструкции, научиться визуализировать результаты и сравнивать модели разной сложности.

Задачи:

- Создать и обучить простой автоэнкодер для сжатия изображений из датасета STL-10.
- Реализовать глубокий автоэнкодер с несколькими скрытыми слоями и посмотреть, как изменяется качество реконструкции.
- Визуализировать и сравнить результаты реконструкции простых и глубоких автоэнкодеров.
- Сделать выводы о том, какую архитектуру предпочтительнее использовать для конкретных задач сжатия изображений.

Ожидаемые результаты:

- Студенты смогут создавать и обучать простые и глубокие автоэнкодеры.
- Поймут, как сложность архитектуры влияет на качество восстановления изображений.
- Смогут визуализировать процессы сжатия и восстановления.
- Будут уметь формулировать обоснованные предположения и рекомендации по улучшению моделей.

Инструменты и библиотеки:

- Язык программирования Python
- Jupyter Notebook / Google Colab
- Библиотеки TensorFlow/Keras, Matplotlib

Ход работы:

Шаг 1. Импорт необходимых библиотек

```
import numpy as np
from tensorflow.keras.datasets import stl10
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Input, Dense, Flatten, Reshape
from tensorflow.keras.optimizers import Adam
```

```
import matplotlib.pyplot as plt
```

Шаг 2. Загрузка и подготовка данных

Мы будем использовать датасет STL-10, состоящий из изображений размером 96×96 пикселей.

```
(train_images, _), (_, _) = stl10.load_data()
train_images = train_images.astype('float32') / 255.0
train_images = train_images.reshape((train_images.shape[0], -1)) # преобразуем
изображения в одномерный вектор
```

Шаг 3. Реализация простого автоэнкодера

Создаем простой автоэнкодер с одним скрытым слоем, используем функцию активации ReLU.

...

```
simple_autoencoder.compile(optimizer=Adam(), loss='binary_crossentropy')
```

Шаг 4. Обучение простого автоэнкодера

Тренировка осуществляется на самих же данных (так как это автоэнкодер, целевым значением являются сами же входные данные).

...

Шаг 5. Реализация глубокого автоэнкодера

Теперь реализуем глубокий автоэнкодер с тремя скрытыми слоями.

...

```
deep_autoencoder.compile(optimizer=Adam(), loss='binary_crossentropy')
```

Шаг 6. Обучение глубокого автоэнкодера

Обучаем второй, более сложный автоэнкодер аналогичным способом.

...

Шаг 7. Визуализация результатов реконструкции

Воспользуемся несколькими случайно выбранными изображениями из датасета и сравним результаты реконструкции простым и глубоким автоэнкодерами.

```
test_image_idx = np.random.randint(len(train_images), size=5)
test_images = train_images[test_image_idx]

reconstructions_simple = simple_autoencoder.predict(test_images)
reconstructions_deep = deep_autoencoder.predict(test_images)

fig, axes = plt.subplots(3, len(test_image_idx), figsize=(len(test_image_idx)*2, 6))
for idx, img_id in enumerate(test_image_idx):
    original_img = test_images[idx].reshape(96, 96, 3)
    reconstr_simple = reconstructions_simple[idx].reshape(96, 96, 3)
    reconstr_deep = reconstructions_deep[idx].reshape(96, 96, 3)

    axes[0][idx].imshow(original_img)
    axes[0][idx].set_title('Original')
    axes[1][idx].imshow(reconstr_simple)
    axes[1][idx].set_title('Simple AE')
```

```
axes[2][idx].imshow(reconstr_deep)
axes[2][idx].set_title('Deep AE')
```

```
plt.tight_layout()
plt.show()
```

Шаг 8. Анализ и сравнение моделей

Рассматриваем графики потерь и сравниваем качество реконструкций, делаем выводы о том, какая модель справляется лучше и почему.

```
plt.figure(figsize=(10, 5))
plt.plot(history_simple.history['loss'], label='Simple Autoencoder Train Loss')
plt.plot(history_deep.history['loss'], label='Deep Autoencoder Train Loss')
plt.title('Train Loss Comparison')
plt.xlabel('Epochs')
plt.ylabel('Binary Cross Entropy Loss')
plt.legend()
plt.show()
```

Требования к отчету:

1. **Титульный лист:** Название работы, ФИО студента, номер группы, дата выполнения.
2. **Введение:** Постановка задачи, описание целей и задач лабораторной работы, обоснование выбора датасета STL-10.
3. **Методология:** Краткое изложение теории автоэнкодеров, принципы работы, разница между простым и глубоким автоэнкодером.
4. **Архитектура и реализация:** Подробное описание реализованных моделей, аргументация выбора функций активации, архитектуры и параметров обучения.
5. **Эксперименты и результаты:** Визуализация оригинального изображения и реконструированного образа, сравнение потерь и реконструкция для обеих моделей.
6. **Анализ и выводы:** Сравнительная характеристика качества реконструкции, обсуждение достоинств и недостатков каждой модели, рекомендации по возможным улучшениям.
7. **Приложения:** Исходный код проекта.

Критерии оценки:

- **Отлично:** Оба типа автоэнкодеров успешно реализованы, проведено полное исследование и сравнение качества реконструкции, сделано качественное оформление отчета.
- **Хорошо:** Обе модели реализованы, но анализ результатов неполный или недостаточно детализирован.
- **Удовлетворительно:** Реализована одна из моделей, присутствует минимальная демонстрация результатов.
- **Неудовлетворительно:** Нет реализации одной из моделей или ошибка в основной части лабораторной работы.

Проверяемые компетенции комплексом практических заданий: ML-4.1; MF-3.2; DL-1.1; DL-1.2; DL-1.5; DL-1.6; DL-1.9; FC-5.2

Зачетно-экзаменационные материалы для промежуточной аттестации (экзамен)

Вопросы для подготовки к экзамену

1. Опишите принципы работы Autoencoder'ов и расскажите о сферах их применения.
2. Что такое graph convolutional networks (GCNs)? Приведите пример их использования.
3. Чем отличаются shallow neural networks от глубоких нейронных сетей?
4. Перечислите и поясните методы аугментации данных.
5. Как работает техника bootstrap sampling в создании ансамблей данных?
6. Почему используется регуляризация в нейронных сетях? Приведите два примера регуляризованных методов.
7. В чём состоит цель оптимизации гиперпараметров нейросети?
8. Чем отличается Grid Search от Random Search?
9. Что такое self-organizing maps (SOMs)? Как они используются?
10. Объясните концепцию Expanding Neuron Gas (ENG) и его роль в обучении нейросетей.
11. Что представляют собой radial basis function (RBF) сети и какова область их применения?
12. Как влияет размер батча на процесс обучения нейросети?
13. Перечислите способы снижения вычислительной нагрузки при работе с большими объёмами данных.
14. Назовите методы оценки эффективности нейросети и их достоинства.
15. Опишите различия между ансамблями данных и ансамблем моделей.
16. В чём заключается особенность энтропийных методов оценки качества обучения нейросети?
17. Какие ограничения имеют традиционные нейросети при обработке последовательных данных?
18. В чём суть концепции explainable artificial intelligence (XAI)?
19. Перечислите общие приёмы повышения объяснимости нейросетевых моделей.
20. Какова роль интерпретирующих инструментов вроде SHAP и LIME в анализе нейросетевых моделей?
21. Какой порядок действий необходим при подготовке большого объёма сырых данных перед созданием нейросети?
22. Как выбрать правильную структуру слоя autoencoder'a для конкретного случая?
23. Каково значение размера шага обучения (learning rate) и как правильно подобрать этот параметр?
24. В каком порядке целесообразно выбирать гиперпараметры нейросети?
25. Какие преимущества даёт ансамблирование нейросетей в прогнозировании финансовых показателей?
26. В чём состоят отличия классической линейной регрессии от искусственных нейросетей?
27. Как оценить качество разработанных нейросетей в задаче прогнозирования?
28. Как оценить качество разработанных нейросетей в задаче классификации?
29. Как оценить качество разработанных нейросетей в задаче кластеризации?
30. Как оценить качество разработанных нейросетей в задаче понижения размерности?

Перечень компетенций (части компетенции), проверяемых оценочным средством
ML-4.1; MF-3.2; DL-1.1; DL-1.2; DL-1.5; DL-1.6; DL-1.9; FC-5.2

Практические задания к экзамену

Практическое задание №1: Подготовка и предобработка набора данных

Задача: Подготовьте датасет MNIST для анализа рукописных цифр, включая выполнение следующего:

- Загрузите и визуально изучите исходные данные.
- Нормализуйте пиксели изображений с помощью MinMaxScaler.
- Преобразуйте целевые метки (цифры) в one-hot представление.

Оцениваемые умения: Навык подготовки и нормализации данных, работа с библиотеками Python (pandas, numpy, scikit-learn).

Практическое задание №2: Аугментация изображений

Задача: Создайте расширенную версию датасета Fashion-MNIST путем аугментации (вращением, сдвигом и изменением яркости).

- Примените трансформации изображений.
- Оцените влияние аугментаций на точность модели.

Оцениваемые умения: Умение применять аугментацию изображений с использованием библиотек типа OpenCV или imgaug.

Практическое задание №3: Обучение ансамблей нейронных сетей

Задача: Реализовать bootstrap sampling на датасете Wine Quality и построить ансамбль простых нейросетей.

- Выполните разделение данных методом бутстрэпа.
- Обучите несколько нейросетей на полученных выборках.
- Объедините прогнозы всех моделей в единый итоговый результат.

Оцениваемые умения: Работа с техникой ансамблевого моделирования, понимание принципов работы bootstrapping.

Практическое задание №4: Простое построение нейросети

Задача: Разработайте трёхслойную нейросеть для распознавания цветков датасета Oxford Flowers 102.

- Определите структуру сети (количество слоёв, количество нейронов, типы слоев).
- Настроить обучение и провести оценку качества модели.

Оцениваемые умения: Создание простой архитектуры нейросети, настройка её структуры и тренировочный процесс.

Практическое задание №5: Автоэнкодеры

Задача: Постройте глубокий автоэнкодер для восстановления изображений из датасета Caltech101.

- Реализуйте глубокую нейросеть-энкодер и дешифратор.
- Проверьте качество реконструкции на тестовых изображениях.

Оцениваемые умения: Применение концепции автоэнкодинга, создание глубоких архитектур нейросетей.

Практическое задание №6: Графовые нейросети

Задача: Используя библиотеку Spektral, реализуйте GCN для решения задачи классификации узлов графа PubMed.

- Научитесь представлять графовую структуру данных.
- Разработайте модель графовой сверточной сети и оцените её производительность.

Оцениваемые умения: Понимание основ графовых нейросетей, опыт работы с специализированными библиотеками.

Практическое задание №7: Легковесные нейросети для изображений и текста

Задача: Разработать легкую CNN-модель для классификации изображений собак и кошек из датасета Dogs vs Cats и легкую RNN-модель для бинарной классификации позитивных и негативных обзоров фильмов IMDB Reviews.

- Спроектируйте и обучите две легкие нейросети.
- Оптимизируйте использование ресурсов памяти и вычислительной мощности.

Оцениваемые умения: Опыт построения легких нейросетевых моделей, балансировка производительности и точности.

Практическое задание №8: Выбор гиперпараметров и оптимизации

Задача: Найдите оптимальное значение скорости обучения (learning rate) и размера батча (batch size) для модели классификации на датасете CIFAR-100.

- Использовать разные значения указанных гиперпараметров.
- Исследуйте влияние каждого параметра на скорость и эффективность обучения.

Оцениваемые умения: Экспериментальное исследование влияния гиперпараметров на качество модели.

Практическое задание №9: Гиперпараметрическая оптимизация

Задача: Примените методику случайного поиска (RandomizedSearchCV) для выбора лучших гиперпараметров для простой логистической регрессии на датасете Boston Housing.

- Проведите гипероптимизацию для числа скрытых слоёв, скоростей обучения и размеров батчей.
- Сравните полученные результаты с результатами, достигнутыми при ручной настройке гиперпараметров.

Оцениваемые умения: Использование методов поиска гиперпараметров, применение техник hyperparameter tuning.

Практическое задание №10: Самоорганизующиеся карты Кохонена (SOFM)

Задача: Реализуйте самоорганизующуюся карту Кохонена для визуализации и кластеризации двумерных данных датасета Breast Cancer Wisconsin Diagnostic Dataset.

- Проанализируйте полученный топологический порядок карт.
- Объясните принцип функционирования SOFM и преимущества метода.

Оцениваемые умения: Освоение концепций SOFM, умение интерпретировать результаты кластеризации.

Практическое задание №11: Прогнозирование временных рядов

Задача: Разработайте рекуррентную нейросеть (RNN/LSTM) для прогнозирования недельных объемов продаж конкретного товара в магазине розничной торговли (используя синтетический временной ряд).

- Генерируйте прогноз на следующий период.
- Измеряйте ошибку модели с помощью MAE и RMSE.

Оцениваемые умения: Анализ временных рядов, проектирование и реализация рекуррентных нейросетей.

Практическое задание №12: Управление кредитными рисками

Задача: Постройте ансамбль нейросетей для прогнозирования дефолтов заемщиков на датасете кредитной истории German Credit Data Set.

- Сформируйте ансамблевый классификатор.
- Сделайте выводы относительно эффективности ансамбля перед одиночными моделями.

Оцениваемые умения: Практическое владение техниками ансамблевых моделей и их применением в финансах.

Практическое задание №13: Интерпретация предсказаний моделей

Задача: Изучите внутреннюю работу нейросети для распознавания мошенничества кредитных карточек (датасет Credit Card Fraud Detection).

- Используйте инструменты интерпретации моделей (SHAP, LIME).
- Получите список наиболее важных признаков для классификации мошеннических транзакций.

Оцениваемые умения: Интерпретация сложных моделей машинного обучения, выявление значимых факторов влияющих на принятие решений моделью.

Эти задания позволят проверить знания студентов по различным аспектам проектирования и реализации нейросетевых алгоритмов, обработке данных, выбору гиперпараметров и работе с временными рядами.

Перечень компетенций (части компетенции), проверяемых оценочным средством
ML-4.1; MF-3.2; DL-1.1; DL-1.2; DL-1.5; DL-1.6; DL-1.9; FC-5.2

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 заноситься 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, выбирать правильные метрики).

% 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)**

82–100% правильных ответов

Демонстрация уверенного владения терминологией и методами.

Хорошо (4)

71–81% правильных ответов

Незначительные ошибки в сложных вопросах.

Удовлетворительно (3)

60–70% правильных ответов

Путаница в базовых концепциях (например, путает метрики, разновидности автокодировщиков).

Неудовлетворительно (2)

Менее 60% правильных ответов

Неспособность отличить архитектуры (например, автокодировщик от графовой нейросети).

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%

Оценка	Экзамен (40%)	Практика (40%)	Тест (20%)	Общий %
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 3-го года обучения.
- Наличие доступа к вычислительным ресурсам (GitLab, Google Colab или Yandex DataSphere, JupyterHub).
- Разработаны лабораторные работы;
- Инфраструктура для приёма задач (gitlab, CI/CD) согласована с лабораторными работами.

Цели, задачи и ожидаемые результаты

Цели организации вычислительной инфраструктуры:

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

Задачи преподавателя:

- Организация регистрации студентов в Google Colab и Yandex DataSphere
- Создание учетных записей студентов в gitlab вуза;
- GitLab Runner;
- Работа над шаблонным репозиторием лабораторных работ;
- Написание инструкции для студентов.

Ожидаемые результаты студентов:

- начальное представление о работе в IT инфраструктуре (гит, нейминг).

Порядок реализации

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

Задача №2: Создание учетных записей студентов в gitlab вуза

Задача №3: GitLab Runner.

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 №4: Работа над шаблонным репозиторием лабораторных работ

Этот репозиторий служит основой для всех новых лабораторных заданий и содержит преднастроенную структуру проекта и CI-конфигурации.

Ключевые файлы и их назначение:

.gitlab-ci.yml – основной конфигурационный файл CI/CD. Определяет среду выполнения (образ Docker), команды для сборки.

Dockerfile – файл для запуска докер-образа.

Задача №5: Написание инструкции для студентов

README.md – инструкции к использованию шаблонного репозитория.

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 3-го года обучения.
- Наличие доступа к вычислительным ресурсам (GitLab) и к GPU/CPU.
- 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 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:

- Навыки разработки нейронных сетей на python.
- Умение подбирать архитектуру нейросети и настраивать гиперпараметры в соответствии с задачей.
- Работа с библиотеками (tensorflow, keras, Spektral и т.д.).
- Опыт решения задач классификации, кластеризации, регрессии и понижения размерности.
- Умение рассчитывать метрики
- Умение интерпретировать результаты обучения нейросети.

Порядок реализации

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

1) Определение тем:

- Предобработка данных
- Аугментация данных
- Ансамблирование данных
- Простое построение нейросети
- Автоэнкодеры
- Работа с графовыми нейросетями
- Легковесные нейросети для текста и изображений
- Выбор гиперпараметров и оптимизационные стратегии
- Гиперпараметрическая оптимизация
- Самоорганизующиеся сети
- Прогнозирование временных рядов
- Управление кредитными рисками
- Портфельное распределение активов
- Интерпретация предсказаний моделей

Наименование Лабораторной работы	Содержание Лабораторной работы	Распределение часов
2	3	
Предобработка а данных	Подготовьте датасет CIFAR-10 (или аналогичную базу изображений) для дальнейшей обработки, выполнив следующие шаги: Загрузка и предварительный просмотр данных.	5 часов (2 часа ЛР, 3 часа СРС)

Наименование Лабораторной работы	Содержание Лабораторной работы	Распределение часов
2	3	
	Заполнение пропущенных значений (если присутствуют). Нормализация данных с помощью MinMaxScaler или StandardScaler. Преобразование категориальных переменных в численные представления. Средства: Python библиотеки pandas, numpy, scikit-learn.	
Аугментация данных	Создайте увеличенный датасет изображений из базы CIFAR-10, применяя методы поворота, сдвига и зеркального отражения изображений. Средства: Библиотеки Python OpenCV, imgaug.	5 часов (2 часа ЛР, 3 часа CPC)
Ансамблирование данных	Примените алгоритм Bootstrap Sampling к базам данных Iris и Wine Quality. Далее обучите простую нейросеть на каждом из созданных подмножеств и выполните агрегацию результатов классификаций. Средства: Python библиотека sklearn.model_selection.	5 часов (2 часа ЛР, 3 часа CPC)
Простое построение нейросети	Разработайте трехслойную нейросеть для классификации цветов розеток из датасета Flowers-102 (цветы). Используйте Dense слои и softmax активацию на последнем слое. Средства: Фреймворк TensorFlow/Keras.	5 часов (2 часа ЛР, 3 часа CPC)
Автоэнкодеры	Реализуйте автоэнкодер для сжатия изображения из датасета STL-10 с последующим восстановлением оригинала. Обратите внимание на разницу в качестве реконструкции простого и глубокого автоэнкодера. Средства: TensorFlow/Keras.	5 часов (2 часа ЛР, 3 часа CPC)
Работа с графовыми нейросетями	Примените Graph Convolutional Network (GCN) для классификации узлов на графовом датасете PPI (Protein-Protein Interaction). Средства: Библиотека Spektral	10 часов (4 часа ЛР, 6 часа CPC)
Легковесные нейросети для текста и изображений	Постройте модель RNN для классификации отзывов IMDb Movie Review (тексты фильмов) и свёрточную сеть LeNet для классификации автомобилей из датасета Cars196. Средства: TensorFlow/Keras.	5 часов (2 часа ЛР, 3 часа CPC)
Выбор гиперпараметров и оптимизационные стратегии	Для задачи классификации на датасете CIFAR-100 проведите эксперименты с различными скоростями обучения, функциями активации и параметрами оптимизатора (Adam, SGD, RMSProp). Средства: TensorFlow/Keras.	5 часов (2 часа ЛР, 3 часа CPC)
Гиперпараметрическая оптимизация	Используйте методики grid search и random search для нахождения наилучших гиперпараметров (learning rate, batch size, number of layers) на датасете Kaggle Titanic Survival Prediction. Средства: Scikit-learn GridSearchCV и RandomizedSearchCV.	5 часов (2 часа ЛР, 3 часа CPC)
Самоорганизующиеся сети	Реализуйте SOFM-карту для кластеризации образов букв английского алфавита из датасета EMNIST Letters. Средства: Python, NumPy, TensorFlow/Keras.	5 часов (2 часа ЛР, 3 часа CPC)

Наименование Лабораторной работы	Содержание Лабораторной работы	Распределение часов
2	3	
Прогнозирование временных рядов	Разработайте нейросеть для краткосрочного прогнозирования цен акций крупной российской компании (например, Сбербанк или Газпром), используя датасет финансового временного ряда из ресурса Yahoo Finance. Средства: TensorFlow/Keras, Yahoo Finance API.	5 часов (2 часа ЛР, 3 часа CPC)
Управление кредитными рисками	Используя датасет кредитного риска German Credit Risk, разработайте ансамбль нейросетей для оценки вероятности дефолта заемщика. Средства: TensorFlow/Keras, scikit-learn.	5 часов (2 часа ЛР, 3 часа CPC)
Интерпретация предсказаний моделей	Исследуйте работу моделей нейросетей на двух разных наборах данных (Credit Card Fraud Detection и Image Classification on Flowers) с использованием инструментов SHAP и LIME для выявления ключевых факторов принятия решений. Средства: Библиотеки SHAP, LIME, TensorFlow/Keras.	5 часов (2 часа ЛР, 3 часа CPC)

2) Разработка заданий:

- Пошаговые инструкции.
- Примеры кода.

Контрольные вопросы.

Разработка заданий для лабораторной работы «Автоэнкодеры»

Цель: Освоить принципы работы автоэнкодеров, научиться проектировать и обучать глубокие и простые нейросети для сжатия и восстановления изображений. Изучить влияние архитектуры и числа слоев на качество реконструкции, научиться визуализировать результаты и сравнивать модели разной сложности.

Технологии: Python, фреймворки TensorFlow, Keras.

Длительность: 4 часа (1 час лабораторных работ, 3 часа самостоятельной работы).

Результат: Код на GitHub + ответы на вопросы

2. Пошаговые инструкции

Для наглядности воспользуемся синтетическим датасетом XOR (исключающее ИЛИ), который требует наличия не-линейной границы разделения.

Шаг 1: Установка библиотек

Шаг 2: Загрузка и подготовка данных

Шаг 3: Реализация простого автоэнкодера

Шаг 4: Обучение простого автоэнкодера

Шаг 5: Реализация глубокого автоэнкодера

Шаг 6: Обучение глубокого автоэнкодера

Шаг 7: Визуализация результатов реконструкции

Шаг 8: Анализ и сравнение моделей

3. Примеры кода на Python

Полный скрипт

python

Копировать

Шаг 1. Импортируем необходимые библиотеки

import numpy as np

```

from tensorflow.keras.datasets import stl10
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Input, Dense, Flatten, Reshape
from tensorflow.keras.optimizers import Adam
import matplotlib.pyplot as plt

# Шаг 2. Загружаем и готовим данные
(train_images, _), (_, _) = stl10.load_data() # загружаем только изображения, игнорируя
классы
train_images = train_images.astype('float32') / 255.0 # нормализуем данные
train_images = train_images.reshape((train_images.shape[0], -1)) # превращаем изображения
в одномерный вектор

# Шаг 3. Реализуем простой автоэнкодер
simple_autoencoder = Sequential([
    Input(shape=(train_images.shape[1],)), # Входной слой принимает плоское
изображение
    Dense(units=64, activation='relu'), # Первый скрытый слой уменьшает размерность
до 64
    Dense(units=train_images.shape[1], activation='sigmoid') # Выходной слой
восстанавливает изображение
])

simple_autoencoder.compile(optimizer=Adam(), loss='binary_crossentropy') # компиляция
модели

# Шаг 4. Обучаем простой автоэнкодер
history_simple = simple_autoencoder.fit(
    train_images, # входные данные
    train_images, # целевой выходной сигнал совпадает с входом
    epochs=10, # кол-во эпох обучения
    batch_size=32, # размер мини-пакетов
    shuffle=True, # перемешиваем данные
    validation_split=0.2 # используем 20% данных для проверки
)

# Шаг 5. Реализуем глубокий автоэнкодер
deep_autoencoder = Sequential([
    Input(shape=(train_images.shape[1],)), # Входной слой
    Dense(units=128, activation='relu'), # первый скрытый слой
    Dense(units=64, activation='relu'), # второй скрытый слой
    Dense(units=32, activation='relu'), # третий скрытый слой
    Dense(units=64, activation='relu'), # четвертый скрытый слой
    Dense(units=128, activation='relu'), # пятый скрытый слой
    Dense(units=train_images.shape[1], activation='sigmoid') # выходной слой
])

deep_autoencoder.compile(optimizer=Adam(), loss='binary_crossentropy') # компиляция
модели

# Шаг 6. Обучаем глубокий автоэнкодер
history_deep = deep_autoencoder.fit(

```

```

train_images,          # входные данные
train_images,          # целевой выходной сигнал совпадает с входом
epochs=10,             # кол-во эпох обучения
batch_size=32,         # размер мини-пакетов
shuffle=True,          # перемешиваем данные
validation_split=0.2   # используем 20% данных для проверки
)

# Шаг 7. Визуализируем реконструкцию изображений
test_image_idx = np.random.choice(range(len(train_images)), size=5) # выбираем 5
случайных изображений
test_images = train_images[test_image_idx]

reconstructions_simple = simple_autoencoder.predict(test_images) # получаем
реконструкцию от простого автоэнкодера
reconstructions_deep = deep_autoencoder.predict(test_images)      # получаем
реконструкцию от глубокого автоэнкодера

# Рисуем изображения рядом друг с другом
fig, axes = plt.subplots(3, len(test_image_idx), figsize=(len(test_image_idx)*2, 6))
for idx, img_id in enumerate(test_image_idx):
    original_img = test_images[idx].reshape(96, 96, 3)          # возвращаем форму
изображения
    reconstr_simple = reconstructions_simple[idx].reshape(96, 96, 3) # восстановление формы
для реконструкции
    reconstr_deep = reconstructions_deep[idx].reshape(96, 96, 3)  # восстановление формы
для реконструкции

    axes[0][idx].imshow(original_img)
    axes[0][idx].set_title('Original')
    axes[1][idx].imshow(reconstr_simple)
    axes[1][idx].set_title('Simple AE')
    axes[2][idx].imshow(reconstr_deep)
    axes[2][idx].set_title('Deep AE')

plt.tight_layout()
plt.show()

# Шаг 8. Сравниваем графики потерь
plt.figure(figsize=(10, 5))
plt.plot(history_simple.history['loss'], label='Simple Autoencoder Train Loss')
plt.plot(history_deep.history['loss'], label='Deep Autoencoder Train Loss')
plt.title('Train Loss Comparison')
plt.xlabel('Epochs')
plt.ylabel('Binary Cross Entropy Loss')
plt.legend()
plt.show()

```

4. Контрольные вопросы

1. Теоретические:
 - Что такое автоэнкодер и для чего он нужен?

- Объясните устройство простой (однослойной) и глубокой (многослойной) архитектуры автоэнкодера.
- Какие функции активации чаще всего применяются в автоэнкодерах и почему?
- Зачем нужны скрытые слои в глубоком автоэнкодере?
- Опишите основную идею принципа обратного распространения ошибки в контексте автоэнкодеров.
- Какова роль функции потерь `binary cross entropy` в автоэнкодерах?
- Расскажите, как устроен этап обучения автоэнкодера?
- Можно ли использовать автоэнкодер для работы с черно-белыми изображениями? Если да, то каким образом?
- Есть ли смысл увеличивать число скрытых слоёв бесконечно? Аргументируйте ваш ответ.
- Чем обусловлена важность правильного выбора размера скрытого слоя?

2. Практические:

- Какой фрагмент вашего скрипта реализует обучение простого автоэнкодера?
- Покажите строку кода, которая генерирует случайные индексы для отображения тестов изображений.
- Где именно происходит формирование нового изображения из сжатых данных в вашем коде?
- Дайте название фрагмента кода, который выводит графики сравнения потерь между простым и глубоким автоэнкодером.
- Назовите ключевую команду, которая запускает обучение глубокого автоэнкодера.
- В каком месте вашего кода происходит инициализация глубоких слоев глубокого автоэнкодера?
- Для чего используется функция `reshape` в вашем скрипте? Где конкретно она встречается?
- Сколько эпох использовалось для тренировки моделей в вашем скрипте?
- Почему используется именно бинарная кросс-энтропия (`binary cross entropy`) в качестве функции потерь? Есть ли альтернативы?
- Перечислите команды, используемые для импорта данных и подготовки их к работе в ваших моделях.

3. Аналитические:

- Насколько существенно повлияло количество скрытых слоев на качество реконструкции изображений? Обсудите ваши наблюдения.
- По вашему мнению, в каких ситуациях предпочтительнее использовать простой автоэнкодер, а в каких — глубокий?
- Существует ли взаимосвязь между количеством эпох обучения и достижимой точностью реконструкции? Почему эта связь существует или отсутствует?
- Является ли уменьшение числа эпох решением проблемы долгой тренировки моделей? Всегда ли это эффективно?
- Представьте ситуацию, когда качество реконструкции улучшилось после перехода от простого автоэнкодера к глубокому. Какие факторы могли повлиять на этот эффект?
- Предположим, вы получили неудовлетворительную точность реконструкции даже после многих эпох обучения. Какие гипотезы вы можете выдвинуть о причинах неудачи и как исправить ситуацию?
- Может ли простая модель показывать лучший результат, чем сложная? Обсудите ситуации, когда это возможно.
- Что произойдет, если размер скрытого слоя значительно уменьшить или увеличить?
- Каковы потенциальные недостатки глубокого автоэнкодера по сравнению с простым?»

- Допустимо ли использовать в автоэнкодере больше выходных нейронов, чем входных? Какие последствия это может иметь?

Критерии оценки

Отлично: Полностью выполнен весь ход работы, продемонстрированы качественные реконструкции изображений, представлены выводы и предложение возможных путей улучшения.

Хорошо: Реализованы обе модели, имеются реконструкции, но не хватает тщательного анализа или качественной визуализации.

Удовлетворительно: Реализована хотя бы одна модель, но нет полного анализа результатов или присутствует значительная погрешность в реализации.

Неудовлетворительно: Модель не создана или серьезно нарушена логика выполнения заданий.

3) Подготовка датасетов:

- Подбор открытых данных с Kaggle и paperswithcode
- 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 3-го года обучения,
- Общее время на проект – не более 16 часов на каждого студента.
- Имеется доступ к кейсам промышленных партнеров; есть возможность адаптации кейсов для студентов третьего курса.

Цели, задачи и ожидаемые результаты

Цели организации вычислительной инфраструктуры:

- дать представление о реальных задачах, решаемых с помощью нейронных сетей 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 экзаменационной оценке

Ожидаемые результаты студентов:

- продвинутое представление о реальных задачах, решаемых с помощью нейросетей и возникающих проблемах
- оценка качества разработанных нейросетей с помощью метрик.

Порядок реализации

Задача №1: сбор кейсов индустриальных партнеров

1. Оптимизация кредитного скоринга с использованием нейронных сетей

Описание: Сбербанк активно развивает кредитные продукты и стремится улучшить процесс оценки кредитоспособности клиентов. Задача – разработать нейронную сеть, которая будет анализировать широкий спектр данных (финансовые показатели, поведенческие данные, социальные сети и т.д.) для более точной оценки риска.

Цель: Создать модель, которая сможет предсказывать вероятность дефолта с высокой точностью, используя различные типы данных.

Ожидаемый результат: Нейронная сеть, способная обрабатывать разнородные данные и выдавать точные прогнозы по кредитоспособности клиентов.

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

Кейс: Распознавание групп мошенников в социальной сети

Постановка задачи: Социальная сеть нуждается в инструменте для обнаружения преступных сообществ среди пользователей. Ваша задача — реализовать графовую нейросеть, способную анализировать связи пользователей и выявлять подозрительные группы.

Требования:

- Примените GCN для классификации пользователей на "добросовестных" и "сомнительных".
- Выявите сообщества, обладающие признаками мошенничества.

Кейс: Детектор поддельных фотографий

Постановка задачи: Нужно создать инструмент для быстрого выявления подделок цифровых изображений. Датасет содержит настоящие и фальшивые снимки знаменитостей.

Требования:

- Обучите лёгкую свёрточную нейросеть (LeNet) для детекции подлинности изображения.
- Определите, насколько точно модель способна отличать реальные фотографии от поддельных.

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

1. Применение различных активационных функций

Задача: Используя библиотеку Spektral, реализуйте GCN для решения задачи классификации узлов графа PubMed.

- Научитесь представлять графовую структуру данных.
- Разработайте модель графовой сверточной сети и оцените её производительность.

Оцениваемые умения: Понимание основ графовых нейросетей, опыт работы с специализированными библиотеками.

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:

- подготовить набор данных для проведения тестирования;
- протестировать разработанные нейросети,
- посчитать функцию потерь
- построить ошибки на обучающих и тестовых данных.
- посчитать метрики
- сделать выводы о качестве нейросети

В зависимости от качества теоретического ответа и количества реализованного самостоятельно кода преподаватель выставляет оценку от 3 до 5.

Критерии оценки:

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

Хорошо: Корректный код, построена функция потерь, сделан правильный вывод о качестве работы модели. Ответил не на все теоретические вопросы.

Удовлетворительно: Выполнение кода с ошибками на этапах визуализации модели и сравнения результатов. 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 оценке за экзамен
Выполнено в РПД, п 4.2

Порядок проверки корректности

Чек-лист для проверки лабораторных работ:

- Набор кейсов промышленных партнеров – 13 шт;
- Набор кейсов преподавателей практиков и лабораторий ВУЗа – 14 шт;
- Набор ТЗ в количестве 14 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. Кохонен, Т. Самоорганизующиеся карты : учебное пособие / Т. Кохонен ; под редакцией Ю. В. Тюменцева ; перевод с английского В. Н. Агеева. – 4-е эл.изд. – Москва : Лаборатория знаний, 2025. – 660 с. – ISBN 978-5-00101-179-8. – Текст : электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/495335> (дата обращения: 21.07.2025). – Режим доступа: для авториз. пользователей.

3. 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.

4. 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 pharmaceutics 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. " ∇^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 и конференции:

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

Конференции А*:

3. <https://openreview.net/forum?id=FMMF1a9ifL>
4. <https://openreview.net/forum?id=ElUrNM9U8c#discussion>

5. <https://openreview.net/forum?id=JoO6mtCLHD>
6. <https://aclanthology.org/2024.findings-emnlp.760/>
7. <https://aclanthology.org/2020.coling-main.588/>
8. https://link.springer.com/chapter/10.1007/978-3-030-72113-8_30
9. https://link.springer.com/chapter/10.1007/978-3-031-42448-9_10
10. <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.prilib.ru/>
8. База данных CSD Кембриджского центра кристаллографических данных (CCDC) <https://www.ccdc.cam.ac.uk/structures/>
9. Springer Journals: <https://link.springer.com/>
10. Springer Journals Archive: <https://link.springer.com/>
11. Nature Journals: <https://www.nature.com/>
12. Springer Nature Protocols and Methods: <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.ucheba.com/>;
17. Законопроект "Об образовании в Российской Федерации". Вопросы и ответы http://xn--273--84d1f.xn--p1ai/voprosy_i_otvety

Собственные электронные образовательные и информационные ресурсы КубГУ

1. Электронный каталог Научной библиотеки КубГУ <http://megapro.kubsu.ru/MegaPro/Web>
2. Электронная библиотека трудов ученых КубГУ <http://megapro.kubsu.ru/MegaPro/UserEntry?Action=ToDb&idb=6>
3. Среда модульного динамического обучения <http://moodle.kubsu.ru>
4. База учебных планов, учебно-методических комплексов, публикаций и конференций <http://infoneeds.kubsu.ru/>
5. Библиотека информационных ресурсов кафедры информационных образовательных технологий <http://mschool.kubsu.ru>;
6. Электронный архив документов КубГУ <http://docspace.kubsu.ru/>
7. Электронные образовательные ресурсы кафедры информационных систем и технологий в образовании КубГУ и научно-методического журнала "ШКОЛЬНЫЕ ГОДЫ" <http://icdau.kubsu.ru/>

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

По дисциплине «Нейросетевые технологии» предусмотрено проведение лекционных занятий, на которых даётся систематизированное представление о некоторых архитектурах нейронных сетей (автокодировщики и их разновидности, графовые нейросети, Expanding Neuron Gas, Skip-gram, CBOW). Студенты знакомятся с этапами подготовки данных для нейросетевого моделирования, генетическими алгоритмами, методами ансамблирования данных (бэггинг, бутстрэп, bootstrapping, stacking, blending). Особое внимание уделяется метрикам оценки качества нейронных сетей в задачах регрессии, классификации, кластеризации и понижения размерности. Важными разделами также являются применение нейросетевых технологий в финансах, а также интерпретация результатов и повышение доверительности AI-моделей.

Лабораторные занятия курса посвящены практическому освоению разработки автокодировщиков, графовых нейросетей и неглубоких нейросетей для работы с текстом и изображениями, ансамблированию данных и прогнозированию временных рядов. На занятиях студенты учатся использовать фреймворки tensorflow и kers, а также библиотеку Spektral. Студенты работают с реальными датасетами, имеющимися в открытом доступе, учатся интерпретации предсказаний нейросети с помощью методов SHAP и LIME. После каждого лабораторного занятия предлагаются задания для самостоятельного закрепления материала - реализация отдельных этапов разработки нейросети или модификация рассмотренных алгоритмов.

При самостоятельной работе студентам необходимо изучать рекомендованную литературу (учебники, научные статьи, документацию библиотек) для глубокого понимания теоретических основ нейросетей. Выполняя проектные задания, студент должен уметь: формулировать задачу (классификация, регрессия, кластеризация, понижение размерности); подбирать и преобразовывать данные; выбирать и реализовывать подходящую архитектуру; интерпретировать результаты работы нейросети. Особое внимание уделяется навыкам подбора итоговых метрик.

Важнейшим компонентом курса является самостоятельная проектная работа, в ходе которой студент разрабатывает законченное приложение. Такой проект позволяет закрепить навыки проектирования и реализации комплексных решений в области нейросетевого моделирования.

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.

Подход, определяющий установление соответствия кейсов ИП и УГТ (5-7), позволяет четко соотносить этапы развития технологии с вовлеченностью партнера и снижать риски при переходе от лабораторных испытаний к промышленному внедрению.

Кейсы ПАО «Сбербанк»

1. Оптимизация кредитного скоринга с использованием нейронных сетей

Описание: Сбербанк активно развивает кредитные продукты и стремится улучшить процесс оценки кредитоспособности клиентов. Задача – разработать нейронную сеть, которая будет анализировать широкий спектр данных (финансовые показатели, поведенческие данные, социальные сети и т.д.) для более точной оценки риска.

Цель: Создать модель, которая сможет предсказывать вероятность дефолта с высокой точностью, используя различные типы данных.

Ожидаемый результат: Нейронная сеть, способная обрабатывать разнородные данные и выдавать точные прогнозы по кредитоспособности клиентов.

2. Обнаружение мошеннических действий

Описание: В Сбербанке ежедневно обрабатывается огромное количество транзакций, и важно выявлять аномалии и подозрительные операции для предотвращения мошенничества. Задача – разработать модель, которая будет анализировать транзакционные данные и выявлять аномалии.

Цель: Создать модель, которая сможет автоматически выявлять аномальные транзакции и предупреждать о возможных мошеннических действиях.

Ожидаемый результат: Нейронная сеть, способная анализировать транзакционные данные и выявлять аномалии с высокой точностью.

3. Оптимизация работы отделений банка

Описание: Сбербанк имеет множество отделений, и важно оптимизировать их работу, улучшая распределение ресурсов и сокращая время ожидания клиентов. Задача – разработать модель, которая будет анализировать данные о работе отделений и улучшать их взаимодействие с клиентами.

Цель: Создать модель, которая сможет обучаться на основе данных о работе отделений и улучшать их взаимодействие с клиентами.

Ожидаемый результат: Нейронная сеть, способная оптимизировать работу отделений банка, за счет кластеризации (например, с помощью самоорганизующейся карты Кохонена), что повысит удовлетворенность клиентов и снизит нагрузку на сотрудников.

4. Модель анализа инвестиционной привлекательности малого бизнеса

Описание: Банк активно развивает кредитование и инвестиционные инструменты для малого и среднего предпринимательства (МСП). Требуется создать модель, которая на основе открытых и банковских данных (выручка, расходы, тип деятельности, отзывы, онлайн-активность) оценивает инвестиционную привлекательность МСП.

Цель: Разработать систему рейтинговой оценки компаний малого бизнеса с возможностью визуализации факторов и динамики показателей.

Ожидаемый результат: Модель, присваивающая компании инвестиционный рейтинг (например, А–Е), объясняющая ключевые параметры и дающая рекомендации для инвестора.

5. Индивидуальная оценка кредитоспособности клиента на основе поведенческих данных

Описание: Современный кредитный скоринг выходит за рамки финансовых данных. Необходимо исследовать, как поведенческие и цифровые следы (частота входа в мобильный банк, способы оплаты, география, время отклика) влияют на персональную оценку риска.

Цель: Разработать нейросеть, оценивающую вероятность дефолта по нестандартным поведенческим признакам.

Ожидаемый результат: Прототип скоринговой модели, которая, помимо стандартных данных, учитывает цифровой профиль клиента и объясняет решения (SHAP, LIME и др.).

6. Оптимизация логистических цепочек доставки услуг с помощью графовых нейросетей

Описание:

Сбер активно развивает сервисы цифровой экономики, такие как доставка еды, посылок и медицинских препаратов. Логистика играет важную роль в успехе бизнеса, особенно учитывая разнообразие регионов обслуживания и специфику инфраструктуры. Решается задача оптимизации маршрута движения курьеров, исходя из текущего местоположения клиентов, трафика и погодных условий.

Цель:

Оптимизация логистических маршрутов для служб доставки с использованием графовых нейросетей, учитывающих условия окружающей среды и состояние дорог.

Ожидаемый результат:

Минимизация затрат на доставку, сокращение сроков исполнения заказов, повышение удовлетворённости клиентов качеством сервиса.

7. Клиентские сегменты для маркетинговых кампаний с использованием автокодировщика

Описание:

Перед Сбером стоит задача сегментировать свою огромную клиентскую базу таким образом, чтобы каждый сегмент соответствовал определенной демографической группе или типу потребительского поведения. Традиционные методы сегментирования становятся неэффективными ввиду огромного объема данных и динамично меняющихся предпочтений потребителей. Для решения этой задачи предлагается использовать автокодировщик для извлечения значимых характеристик клиентов.

Цель:

Создание эффективных клиентских сегментов на основе глубоких представлений, извлекаемых автокодировщиком, для точной таргетированной рекламы и маркетинга.

Ожидаемый результат:

Повышение эффективности рекламных кампаний, рост конверсии маркетинговых сообщений и общее удовлетворение клиентов благодаря индивидуальным подходам к коммуникации.

Кейсы от «АВАЛАБ»

1. Модель прогнозирования сроков сдачи объектов на основе табличных данных

Описание: Девелоперская компания ведёт аналитический архив по срокам строительства. С помощью нейросетевой модели можно прогнозировать вероятность отклонения от графика сдачи.

Цель: Разработать модель, которая по текущему статусу объекта (отчёт СМР) оценивает риски задержек.

Ожидаемый результат: Прототип, который показывает вероятность отклонений и даёт текстовые пояснения о том, на каком этапе произошла задержка сроков.

2. Прогнозирование сроков строительства с использованием нейронных сетей

Описание: Строительная компания сталкивается с необходимостью точного прогнозирования сроков завершения строительных проектов. Задача – разработать модель, которая будет анализировать данные о предыдущих проектах и прогнозировать сроки завершения текущих проектов.

Цель: Создать модель, которая сможет предсказывать сроки завершения проектов с высокой точностью, учитывая различные факторы, такие как погодные условия, доступность ресурсов и другие.

Ожидаемый результат: Нейронная сеть, способная анализировать данные и выдавать точные прогнозы по срокам завершения проектов.

3. Анализ качества строительных материалов с использованием нейронных сетей

Описание: Строительная компания использует различные строительные материалы, и важно контролировать их качество. Задача – разработать модель, которая будет анализировать данные о материалах и прогнозировать их качество.

Цель: Создать модель, которая сможет анализировать данные о материалах и предсказывать их качество, что поможет избежать использования некачественных материалов.

Ожидаемый результат: Нейронная сеть, способная анализировать данные и выдавать прогнозы по качеству строительных материалов.

4. Ранняя диагностика технического износа оборудования с помощью автокодировщика

Описание:

Строительство требует эксплуатации дорогостоящего оборудования: экскаваторов, бульдозеров, бетономешалок и прочих агрегатов. Своевременное выявление технических неисправностей снижает затраты на ремонт и повышает надежность работы парка техники. Текущие методы профилактического осмотра устарели и требуют значительных ресурсов. Важно найти новый подход к диагностике технического состояния техники.

Цель:

Применение автокодировщика для анализа больших объемов данных датчиков, собираемых с строительного оборудования, и раннего выявления поломок.

Ожидаемый результат:

Снижение стоимости ремонтов, продление срока службы оборудования, повышение надежности и экономичности производства.

5. Планирование снабжения материалами с использованием графовых нейросетей

Описание:

Логистика поставок стройматериалов имеет решающее значение для успешной организации строительно-монтажных работ. Компании приходится решать сложную задачу координации закупок и поставки материалов на объекты, принимая во внимание сроки строительства, доступность поставщиков и стоимость транспортировки. Процесс

снабжения становится сложнее из-за географически разбросанной сети объектов и нестабильных рыночных условий.

Цель:

Развитие инструмента планирования закупок и поставок на основе графовых нейросетей, который обеспечит оптимальный баланс между сроками строительства и стоимостью материалов.

Ожидаемый результат:

Эффективное управление запасами и логистическими потоками, сокращение расходов на закупки и транспортировку, повышение рентабельности проектов.

6. Оптимизация управления ресурсами на стройплощадке

Описание: Девелоперская компания управляет множеством ресурсов на стройплощадке, включая материалы, оборудование и персонал. Задача – разработать модель, которая будет оптимизировать управление ресурсами, улучшая их использование и сокращая простои.

Цель: Создать модель, которая сможет обучаться на основе данных о ресурсах и улучшать их распределение и использование.

Ожидаемый результат: Нейросеть, способная оптимизировать управление ресурсами на стройплощадке за счет кластеризации (например, с помощью самоорганизующейся карты Кохонена), что повысит эффективность работы компании и снизит затраты.

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

7.1 Перечень информационно-коммуникационных технологий

1. Облачные платформы и сервисы

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

Kaggle – платформа для работы с датасетами и соревнований по ML

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

2. Системы управления версиями и коллаборации

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

3. Инструменты для работы с данными

DVC (Data Version Control) – управление версиями данных

4. Система управления обучением

Moodle – сдача работ

7.2 Перечень лицензионного и свободно распространяемого программного обеспечения

1. Лицензионное ПО

VSCoде – IDE для Python (свободно распространяемое)

LibreOffice – оформление отчетов (свободно распространяемое)

2. Свободное ПО (Open Source)

Фреймворки и библиотеки для нейросетей:

Scikit-learn – разработка простых нейросетей

TensorFlow/Keras – разработка нейросетей

Spektral – графовые нейросети

Инструменты для визуализации и обработки данных:

OpenCV/ imgaug – для работы с изображениями

Pandas – предобработка данных

Numpy – работа с векторами и матрицами

Streamlit/Gradio – создание веб-интерфейсов для моделей

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
		Аренда публичного IP	1		Шт
2	Виртуальная машина с GPU	Виртуальная машина с GPU NVIDIA® Tesla® V100 2 GPU 8 vCPU 128 Гб RAM	1	1	Шт
		ОС Ubuntu_24.04	1		Шт
		Системный диск SSD	1		Шт
			2000		Гб
		Диск SSD	1		Шт
			4096		Гб
		Диск SSD	1		Шт
			4096		Гб
3	K8S	Аренда публичного IP	1	1	Шт
		Master node 8 vCPU 16 RAM	1		Шт
		Worker node 10% доля 4 vCPU 32 RAM	5		Шт
		Worker node SSD-NVME	64		Гб
		Аренда публичного IP	1		Шт
4	ML Inference Instance Type GPU	Время работы в месяц	40	1	Ч

Инстанс 8 x NVIDIA® H100 NVLink PCIe 160 vCPU 1520 GB RAM	1	Шт
Количество запросов к ML- моделям	1	Млн. Шт
Кэш ML-моделей	160	Гб

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.