

Федеральное государственное образовательное бюджетное учреждение высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)

Кафедра искусственного интеллекта
Факультет информационных технологий и анализа больших данных

Документ подписан усиленной неквалифицированной электронной подписью.

Организация: «ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»

Сертификат: zDpCq/7kznO8jNhHFmr3aOt4rWAK5D3W

Утверждено: Проректор по учебной и методической работе, Е.А. Каменева

Дата: 29.05.2026 г.

С.В. Макрушин , З.Х. Калажоков

Основы глубокого обучения

Рабочая программа дисциплины

для студентов, обучающихся по направлению подготовки

09.03.03 - Прикладная информатика,

Образовательная программа «Прикладные информационные системы в экономике и
финансах»

Профиль:

«Прикладные информационные системы в экономике и финансах»

Рекомендовано

Факультет информационных технологий и анализа больших данных

(протокол № 08 от 19.05.2026 г.)

Одобрено

Кафедра искусственного интеллекта

(протокол № 9 от 19.06.2026 г.)



Содержание

1. Наименование дисциплины	3
2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине	3
3. Место дисциплины в структуре образовательной программы	6
4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся	6
5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий	7
5.1. Содержание дисциплины	7
5.2. Учебно – тематический план	9
5.3. Содержание семинаров, практических занятий	10
6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	13
6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы	13
6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю	17
7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	18
8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	3
9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	6
10. Методические указания для обучающихся по освоению дисциплины	29
11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)	31
12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	31



1. Наименование дисциплины

Основы глубокого обучения

2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
ПКП-1	Способен разрабатывать и отлаживать программный код	Выполняет формализацию и алгоритмизацию поставленных задач для разработки программного кода	Знать: способы формализации задач (формулы, таблицы, графы, блок-схемы), виды и свойства алгоритмов, нотации описания алгоритмов (псевдокод, блок-схемы по ГОСТ, UML), понятия входных/выходных данных и крайних условий, основы оценки сложности алгоритмов (О-нотация) Уметь: анализировать ТЗ и выделять требования, переводить задачу на формальный язык, декомпозировать задачу на подзадачи, составлять алгоритм в виде блок-схемы или псевдокода, выполнять трассировку алгоритма на тестовых примерах, обосновывать выбор структур данных и алгоритмов
		Оформляет программный код в соответствии установленными требованиями	Знать: стандарты оформления кода (PEP 8), правила именования (camelCase, snake_case, PascalCase), требования к форматированию (отступы, длина строки, пробелы), правила комментирования (docstring, TODO), структуру модулей и пакетов, назначение статических анализаторов (линтеров)



Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
			Уметь: писать код в заданном стиле, использовать автоматические форматтеры, грамотно комментировать сложные участки, выбирать осмысленные имена переменных и функций, выделять повторяющийся код в методы, проверять код
ПКП-6	Способен участвовать в промышленной разработке программного обеспечения	Работает в соответствии с промышленными методологиями разработки	Знать: основные промышленные методологии (Waterfall, Agile, Scrum, Kanban), их принципы, этапы и артефакты, роли в команде (владелец продукта, Scrum-мастер, разработчик), понятия спринта, бэклога, ежедневных встреч, ретроспективы, доски задач, различие между итеративной и инкрементальной разработкой Уметь: работать в рамках выбранной методологии, участвовать в планировании спринта/итерации, оценивать трудоёмкость задач (story points, часы), вести свой бэклог задач, соблюдать циклы разработки (code review, тестирование, интеграция), адаптироваться под изменения требований в Agile-среде
		Применяет инструменты поддержки методологий разработки и использует навыки ведения документации в соответствии с требованиями методологии	Знать: инструменты для управления задачами (Jira, Trello, YouTrack, Azure DevOps), системы контроля версий (Git) и правила работы с ветками (Git Flow, GitHub Flow), инструменты для ведения документации (Confluence, Wiki, Markdown), требования к составлению технической документации (пользовательская, API, внутренняя), понятия CI/CD и связь с методологиями Уметь: создавать и обновлять задачи в трекере, привязыв-



Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
			вать коммиты к задачам, оформлять документацию по шаблонам методологии (README, инструкции по сборке/развёртыванию), работать с вики или Confluence, оформлять результаты ретроспективы и code review в письменном виде, использовать Git с соблюдением принятого в команде workflow



3. Место дисциплины в структуре образовательной программы

Дисциплина «Основы глубокого обучения» относится к «Модулю "Технологии машинного обучения"».

4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

Заочная форма обучения

Вид учебной работы по дисциплине	Всего (в з/е и часах)	Семестр 8 (в часах)
Общая трудоемкость дисциплины	3/108	108
Контактная работа – Аудиторные занятия	10	10
<i>Лекции</i>	<i>2</i>	<i>2</i>
<i>Семинары, практические занятия</i>	<i>8</i>	<i>8</i>
<i>Самостоятельная работа</i>	<i>98</i>	<i>98</i>
Вид текущего контроля	Контрольная работа	Контрольная работа
Вид промежуточной аттестации	Зачет	Зачет



5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Введение в нейронные сети

Перцептрон. Взвешенная сумма. Функции активации Sigmoid, Tanh, ReLU. Анатомия нейронной сети. Распространение сигнала по нейронной сети. Обратное распространение ошибки. Описание обратного распространения весов с помощью матричной алгебры.

Тема 2. Обучение глубоких сетей

Функции стоимости: квадратичная функция стоимости, перекрестная энтропия. Насыщенные нейроны. Корректировка весовых коэффициентов. Оптимизация обучения методом минимизации стоимости. Градиентный спуск. Начальная инициализация весов. Распределения Ксавье-Глоро. Борьба с переобучением. Регуляризация L1, L2. Прореживание. Обогащение данных. Затухающие градиенты.

Тема 3. Глубокое обучение в технологиях компьютерного зрения

Сверточные нейронные сети. Операция свертывания. Max-pooling. Фильтры. Предварительно обученные сверточные сети. Многомерная свертка. Сети LeNet-5, AlexNet и VGGNet. Компьютерное зрение.

Тема 4. Глубокое обучение для текста и последовательностей

Работа с текстовыми данными. Обработка естественного языка. Прямое кодирование слов и символов. Лексемизация, стемминг. Обработка n-грамм. Векторное представление слов. Алгоритм word2vec. Площадь под кривой ROC. Оценка AUC. Матрица ошибок.

Тема 5. Рекуррентные нейронные сети

Структура рекуррентной сети (RNN). Обучение рекуррентной сети. Длительная краткосрочная память (LSTM).



Тема 6. Механизм внимания (Attention) и трансформеры

Self-attention, multi-head attention, позиционное кодирование, архитектура трансформера (encoder-decoder), сравнение с RNN/CNN.

Тема 7. Генеративные модели

Автокодировщики (VAE), генеративно-состязательные сети (GAN), диффузионные модели (краткий обзор), примеры генерации изображений.

Тема 8. Обучение с подкреплением

Обучение с подкреплением. Марковские процессы принятия решений. Сети Q-обучения. Определение агента DQN.



5.2. Учебно – тематический план

Заочная форма обучения

п/ п	Наименование тем (разделов) дисциплины	Трудоемкость в часах				
		Всего	Контактная работа - Аудиторная работа			Самостоятельная работа
			Общая, в т.ч.:	Лекции	Семинары, практические занятия	
1	Введение в нейронные сети	14	2	1	1	12
2	Обучение глубоких сетей	14	2	1	1	12
3	Глубокое обучение в технологиях компьютерного зрения	13	1	0	1	12
4	Глубокое обучение для текста и последовательностей	13	1	0	1	12
5	Рекуррентные нейронные сети	13	1	0	1	12
6	Механизм внимания (Attention) и трансформеры	13	1	0	1	12
7	Генеративные модели	13	1	0	1	12
8	Обучение с подкреплением	15	1	0	1	14
В целом по дисциплине		108	10	2	8	98



5.3. Содержание семинаров, практических занятий

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
Введение в нейронные сети	Понятие нейронной сети, ее математическая модель Обратное распространение ошибки. Описание обратного распространения ошибок с помощью матричной алгебры.	Интерактивная форма. Практикум по решению задач по тематике занятия в малых группах (2-4 студента) и коллективное обсуждение решений.
Обучение глубоких сетей	Каковы практические признаки переобучения и как их вовремя заметить по графикам loss и ассигасу на обучающей и валидационной выборках? В каких ситуациях Adam даёт лучшую сходимость, чем классический SGD с моментом, а когда стоит вернуться к SGD? Разбор кейса: loss на обучении падает, а на валидации растёт с первой эпохи — в чём возможные причины и какие гиперпараметры проверить в первую очередь?	Интерактивная форма. Практикум по решению задач по тематике занятия в малых группах (2-4 студента) и коллективное обсуждение решений.
Глубокое обучение в технологиях компьютерного зрения	Как выбрать между использованием предобученной сети (transfer learning) и обучением с нуля, если у вас всего 500 размеченных изображений? Сравнение на примерах, когда эффективнее fine-tuning всей сети, а когда — только классификатора (заморозка свёрточных слоёв). Почему в современных архитектурах всё чаще отказываются от max-pooling в пользу свёрток с шагом 2?	Интерактивная форма. Практикум по решению задач по тематике занятия в малых группах (2-4 студента) и коллективное обсуждение решений.
Глубокое обучение для текста и	В чём сильные и слабые стороны one-hot encoding по сравнению с плотными эмбедингами (word2vec, GloVE)	Интерактивная форма. Практикум по решению задач по тематике занятия в малых группах (2-4 студента) и



Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
последовательностей	для задачи классификации коротких сообщений (например, твитов)? Как интерпретировать матрицу ошибок и почему ассигасу может обманывать при сильном дисбалансе классов? Разбор метрики AUC-ROC: как её значение 0,5, 0,8 и 0,95 трактовать на примере задачи спам-фильтра.	коллективное обсуждение решений.
Рекуррентные нейронные сети	Почему LSTM эффективнее классической RNN для предсказания временных рядов с горизонтом в сотни шагов, и можно ли обойтись без LSTM используя Feature Engineering? Сравнение двунаправленных (BiLSTM) и однонаправленных LSTM на задаче определения тональности отзыва: в каких сценариях BiLSTM не применим в реальном времени? Как затухающий градиент убивает обучение простой RNN, покажите на маленькой сети с 10 слоями.	Интерактивная форма. Практикум по решению задач по тематике занятия в малых группах (2-4 студента) и коллективное обсуждение решений.
Механизм внимания (Attention) и трансформеры	Почему self-attention имеет вычислительную сложность $O(n^2)$ и для последовательностей какой длины это становится проблемой даже на современных GPU? Сравнение трансформера и LSTM на задаче машинного перевода: за счёт чего трансформер обучается быстрее, но требует больше памяти? Что произойдёт с выходом энкодера, если убрать позиционное кодирование — будет ли порядок слов вообще учитываться?	Интерактивная форма. Практикум по решению задач по тематике занятия в малых группах (2-4 студента) и коллективное обсуждение решений.



Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
Генеративные модели	В чём разница между вариационным автокодировщиком и обычным: почему обычный автокодировщик не умеет генерировать новые цифры, а только сжимать и восстанавливать? Какие артефакты на изображениях выдают обученную GAN (например, нечёткость, отсутствие мелких деталей) и как на это влияет коллапс мод? Разбор диффузионной модели: почему обратный процесс требует гораздо больше шагов (тысячи против десятков у GAN), но даёт более реалистичные результаты.	Интерактивная форма. Практикум по решению задач по тематике занятия в малых группах (2-4 студента) и коллективное обсуждение решений.
Обучение с подкреплением	Как сформулировать задачу рекомендации фильмов как марковский процесс принятия решений (MDP): что будет состояниями, действиями и наградами? Почему DQN нуждается в replay buffer и целевой сети, тогда как простой Q-learning без нейросетей может обновляться на каждом шаге? Дилемма exploration–exploitation на примере: как агенту научиться играть в шахматы, если он никогда не пробует проигрышные ходы?	Интерактивная форма. Практикум по решению задач по тематике занятия в малых группах (2-4 студента) и коллективное обсуждение решений.



6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Введение в нейронные сети	Объясните своими словами, как сигнал проходит от входов до выходов через взвешенную сумму и функцию активации. В чём ключевое отличие обратного распространения ошибки от прямого распространения сигнала? Почему сигмоида и $\tanh$ считаются «насыщаемыми» функциями, а ReLU — нет?	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий.
Обучение глубоких сетей	Почему перекрёстная энтропия лучше квадратичной стоимости для задач классификации с насыщаемыми нейронами? Как начальная инициализация весов по Ксавье-Глоро помогает избежать затухающих градиентов? Чем регуляризация L1 отличается от L2 и прореживания (dropout) по влиянию на веса?	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий.
Глубокое обучение в технологиях компьютерного зрения	Зачем в свёрточных сетях нужны операции свёртки и max-pooling, и что произойдёт, если убрать pooling полностью? Чем архитектура LeNet-5 отличается от AlexNet, а AlexNet — от VGGNet? Что означает «предварительно обученная свёрточная сеть» и как её применить к новой задаче?	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий.
Глубокое обучение для текста и	В чём разница между лексемизацией (tokenization) и стеммингом, и когда что применять? Как алгоритм	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий.



Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
последовательностей	word2vec обучает векторные представления слов без ручной разметки семантики? Что показывает матрица ошибок и почему AUC-ROC считается лучшей метрикой, чем просто точность для несбалансированных классов?	заданий.
Рекуррентные нейронные сети	Почему классическая RNN не способна запоминать зависимости на много шагов назад, а LSTM может? Какие три «вентиля» в LSTM управляют потоком информации, и за что отвечает каждый? Чем обучение RNN (через развёртывание во времени) отличается от обучения обычного многослойного перцептрона?	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий.
Механизм внимания (Attention) и трансформеры	В чём фундаментальное отличие self-attention от рекуррентной обработки последовательности? Как multi-head attention позволяет сети «смотреть» на входные данные с разных сторон одновременно? Почему трансформеру нужно позиционное кодирование, ведь RNN и CNN в нём не нуждаются?	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий.
Генеративные модели	Чем вариационный автокодировщик (VAE) отличается от простого автокодировщика и почему первый умеет генерировать новые данные, а второй — нет? Как в GAN генератор и дискриминатор «состязаются» друг с другом и что происходит при коллапсе мод (mode collapse)? Кратко: в чём идея диффузионных моделей и почему они сейчас популярны?	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий.



Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Обучение с подкреплением	Что такое марковский процесс принятия решений (MDP) и какие пять компонентов его описывают? Как Q-обучение заменяет оценку будущей награды на функцию ценности действия без знания модели среды? Зачем DQN использует replay buffer и отдельную целевую сеть (target network), а не обучается на каждом шаге сразу?	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий.



6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю

Примерные вопросы к контрольной работе

1. Геометрическая интерпретация работы нейрона: как веса и смещение (bias) определяют положение разделяющей гиперплоскости.
2. Проблема затухающих и взрывных градиентов: причины возникновения и влияние выбора функции активации на эту проблему.
3. Переобучение (Overfitting): визуальные признаки на графиках обучения и фундаментальные причины явления.
4. Методы регуляризации: принцип работы Dropout, L1 и L2-регуляризации.
5. Пакетная нормализация (Batch Normalization): зачем она нужна и как ускоряет сходимость сети.
6. Механизмы «памяти» в архитектурах LSTM и GRU: решение проблемы кратковременной памяти классических RNN.
7. Динамика скорости обучения (Learning Rate): риски слишком высокого и слишком низкого шага обучения, использование Schedulers.
8. Размер батча (Batch Size): компромисс между точностью оценки градиента, скоростью вычислений и обобщающей способностью модели.
9. Инвариантность сверточных слоев: почему CNN устойчивы к сдвигам и искажениям объектов на изображениях.
10. Понятие рецептивного поля (Receptive Field): как увеличивается область обзора нейрона при движении вглубь сети.
11. Проблема несбалансированных данных: стратегии обучения, когда один класс представлен значительно шире другого.
12. Аугментация данных: способы искусственного расширения обучающей выборки и её влияние на инвариантность.
13. Метрики классификации: почему Accuracy может быть бесполезна и когда использовать Precision, Recall и F1-меру.
14. Стратегии Fine-tuning: как правильно дообучать веса при использовании Transfer Learning в зависимости от размера данных.
15. Адаптивные оптимизаторы (Adam, RMSprop, Adagrad): в чем их отличие от классического стохастического градиентного спуска (SGD).
16. Метод моментов (Momentum): физическая аналогия и помощь в преодолении локальных минимумов и седловых точек.
17. Инициализация весов (Xavier, He): математическое обоснование выбора начальных значений для сохранения дисперсии сигналов.



18. Динамический вычислительный граф в PyTorch: как формируется цепочка производных во время прямого прохода.
19. Концепция автоматического дифференцирования (Autograd): как библиотека вычисляет градиенты сложных композитных функций.
20. Этап валидации vs Тестирование: почему нельзя подбирать гиперпараметры, опираясь только на тестовую выборку.
21. Лосс-функции (Loss Functions): обоснование выбора Cross-Entropy для классификации и MSE для регрессии.
22. Эвристические методы поиска: основы генетических алгоритмов и имитации отжига в контексте настройки весов.
23. Архитектурные "бутылочные горлышки" (Bottlenecks): как сужение слоев влияет на сжатие и извлечение ключевых признаков.

Дополнительная информация

Критерии балльной оценки различных форм текущего контроля успеваемости содержатся в соответствующих методических рекомендациях Кафедры искусственного интеллекта Факультета информационных технологий и анализа больших данных.



7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Перечень компетенций с указанием индикаторов их достижения в процессе освоения образовательной программы содержится в разделе 2. «Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине».

Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний

Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
ПКП-1. Способен разрабатывать и отлаживать программный код	Выполняет формализацию и алгоритмизацию поставленных задач для разработки программного кода	Знать: способы формализации задач (формулы, таблицы, графы, блок-схемы), виды и свойства алгоритмов, нотации описания алгоритмов (псевдокод, блок-схемы по ГОСТ, UML), понятия входных/выходных данных и краевых условий, основы оценки	Разработать автокодировщик для предоставленных данных, провести обучение сети.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		сложности алгоритмов (О-нотация) Уметь: анализировать ТЗ и выделять требования, переводить задачу на формальный язык, деккомпозировать задачу на подзадачи, составлять алгоритм в виде блок-схемы или псевдокода, выполнять трассировку алгоритма на тестовых примерах, обосновывать выбор структур данных и алгоритмов	
	Оформляет программный код в соответствии установленными требованиями	Знать: стандарты оформления кода (РЕР 8), правила именования (camelCase,	Согласно РЕР 8, какая максимальная рекомендуемая длина строки кода? Что делать, если строка объективно должна быть длиннее?



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		snake_case, PascalCase), требования к форматированию (отступы, длина строки, пробелы), правила комментирования (docstring, TODO), структуру модулей и пакетов, назначение статических анализаторов (линтеров) Уметь: писать код в заданном стиле, использовать автоматические форматтеры, грамотно комментировать сложные участки, выбирать осмысленные имена переменных и функ-	



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		ций, выделять повторяющийся код в методы, проверять код	
ПКП-6. Способен участвовать в промышленной разработке программного обеспечения	Работает в соответствии с промышленными методологиями разработки	Знать: основные промышленные методологии (Waterfall, Agile, Scrum, Kanban), их принципы, этапы и артефакты, роли в команде (владелец продукта, Scrum-мастер, разработчик), понятия спринта, бэклога, ежедневных встреч, ретроспективы, доски задач, различие между итеративной и инкрементальной разработкой Уметь: работать в	Перечислите не менее 3 артефактов Scrum и кратко опишите каждый.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		рамках выбранной методологии, участвовать в планировании спринта/итерации, оценивать трудоёмкость задач (story points, часы), вести свой бэклог задач, соблюдать циклы разработки (code review, тестирование, интеграция), адаптироваться под изменения требований в Agile-среде	
	Применяет инструменты поддержки методологий разработки и использует навыки ведения документации в соответствии с	Знать: инструменты для управления задачами (Jira, Trello, YouTrack, Azure DevOps), системы контроля версий (Git)	Какие из перечисленных инструментов предназначены для управления задачами, а какие — для ведения документации: Jira, Confluence, Trello, YouTrack, Wiki, Azure DevOps? Ответ запишите в виде двух списков.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
	требованиями методологии	и правила работы с ветками (Git Flow, GitHub Flow), инструменты для ведения документации (Confluence, Wiki, Markdown), требования к составлению технической документации (пользовательская, API, внутренняя), понятия CI/CD и связь с методологиями Уметь: создавать и обновлять задачи в трекере, привязывать коммиты к задачам, оформлять документацию по шаблонам методологии	



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		(README, инструкции по сборке/развёртыванию), работать с вики или Confluence, оформлять результаты ретроспективы и code review в письменном виде, использовать Git с соблюдением принятого в команде workflow	



Примеры практико-ориентированных заданий

Задание 1.

Обучите многослойный перцептрон на датасете MNIST (рукописные цифры) с нуля на Python без использования готовых фреймворков глубокого обучения (разрешён только NumPy). Реализуйте прямое распространение, обратное распространение ошибки и оптимизацию методом SGD. Добитесь точности на тестовой выборке не менее 85%.

Задание 2.

Используя предобученную модель ResNet18 из библиотеки PyTorch или TensorFlow, дообучите (fine-tune) её для классификации пород собак на датасете Stanford Dogs (120 пород). Заморозьте первые 10 слоёв, замените последний полносвязный слой и обучите только новые слои. Оцените точность топ-1 и топ-3 на отложенной выборке.

Задание 3.

Постройте модель LSTM для генерации имён пользователей (или названий брендов). Обучите её на списке из 10 000 реальных имён (можно взять открытый датасет). На входе — символ за символом, на выходе — следующий символ. После обучения сгенерируйте 50 новых имён, выбрав температуру softmax = 0,8 для баланса между повторениями и хаосом.

Задание 4.

Реализуйте механизм self-attention вручную на NumPy или PyTorch без использования готовых модулей nn.MultiheadAttention. Возьмите произвольную последовательность из 5 векторов размерности 64, вычислите Query, Key, Value через случайные матрицы весов, затем посчитайте scaled dot-product attention и multi-head attention с 4 головами. Сравните выход с реализацией из фреймворка.

Задание 5.

Обучите простую GAN (генератор и дискриминатор — полносвязные сети) на датасете рукописных цифр MNIST. Генератор должен создавать реалистичные цифры 28×28 пикселей из случайного шума размерности 100. Визуализируйте процесс обучения: каждые 5 эпох сохраняйте сгенерированные изображения и график потерь генератора и дискриминатора. Добейтесь, чтобы сгенерированные цифры были визуально узнаваемы.

Примеры вопросов для подготовки к промежуточной аттестации

Примерные вопросы для подготовки к зачету

1. Понятие искусственных нейронных сетей. Характерные черты ИНС.
2. Важнейшие свойства биологических и искусственных нейросетей.
3. Функции активации.
4. Классификация НС.



5. Задачи, решаемые с помощью НС.
6. Типы обучения нейронных сетей.
7. Обучение НС. Обучение “с учителем”, ”Обучение без учителя”.
8. Недостатки метода обратного распространения ошибки. Модификации алгоритма.
9. Персептрон. Архитектура, методы обучения.
10. Линейно разделимые и неразделимые задачи. Проблема исключающего ИЛИ. Необходимость использования многослойных сетей
11. Структура сети прямого распространения.
12. Многослойный персептрон. Типы слоев и параметры сети.
13. Обучение многослойных нейронных сетей. Методы оптимизации.
14. Алгоритм обратного распространения ошибки.
15. Стохастический градиентный спуск.
16. Задача классификации.
17. Рекуррентные НС. Особенности НС.
18. Глобальные методы оптимизации. Генетический алгоритм. Алгоритм имитации отжига.
19. Проектирование, обучение и моделирование НС с помощью пакета PyTorch.
20. Алгоритмы обучения: градиентные, методы сопряженных градиентов, квазиньютоновские методы.
21. Методы минимизации функционала ошибки при обучении НС.
22. Создание обучающего множества (обучающее, контрольное, тестовое).
23. Функции инициализации весов.
24. Функции обучения сети. Параметры обучения.
25. Моделирование сети.

** Промежуточная аттестация обучающихся проводится в формате тестирования*



8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Гаваев, А. С. Основы систем искусственного интеллекта [Электронный ресурс] : учебное пособие / Гаваев А. С., Лысенко И. В., Чайников Д. А., Губенко А. С. Тюмень : ТИУ, 2024 88 с. Книга из коллекции ТИУ - Информатика <https://e.lanbook.com/book/513645> ISBN 978-5-9961-3402-1. [БИК ID: RU-LAN-BOOK-513645]
2. Антонов, И. В. Прикладное глубокое обучение [Электронный ресурс] : учебное пособие / Антонов И. В., Бруттан Ю. В. Псков : ПсковГУ, 2024 138 с. Книга из коллекции ПсковГУ - Информатика <https://e.lanbook.com/book/494504> ISBN 978-5-00200-228-3. [БИК ID: RU-LAN-BOOK-494504]

Дополнительная литература:

1. Родимков, Ю. А. Компиляторы для глубоких нейросетевых моделей [Электронный ресурс] : учебно-методическое пособие / Родимков Ю. А., Мееров И. Б., Сысоев А. В., Кустикова В. Д. Нижний Новгород : ННГУ им. Н. И. Лобачевского, 2025 20 с. Рекомендовано методической комиссией института ИТММ для студентов ННГУ, обучающихся по направлениям подготовки 01.03.02 «Прикладная математика и информатика», 01.04.02 «Прикладная математика и информатика», 02.03.02 «Фундаментальная информатика и информационные технологии», 02.04.02 «Фундаментальная информатика и информационные технологии» Книга из коллекции ННГУ им. Н. И. Лобачевского - Информатика <https://e.lanbook.com/book/507840>. [БИК ID: RU-LAN-BOOK-507840]
2. Соловьев, В.И. Анализ данных в экономике: Теория вероятностей прикладная статистика обработка и анализ данных в Microsoft Excel : Учебник / В.И. Соловьев Электрон. дан. Москва : КноРус, 2023 497 с. Режим доступа: book.ru Internet access <https://book.ru/book/946789> ISBN 978-5-406-10701-0. [БИК ID: RU\bookru\bibl\946789]
3. Коротеев, М.В. Основы машинного обучения на Python : Учебник / М.В. Коротеев Электрон. дан. Москва : КноРус, 2024 431 с. Режим доступа: book.ru Internet access <https://book.ru/book/952751> ISBN 978-5-406-12673-8. [БИК ID: RU\bookru\bibl\952751]



4. Никитин, П.В. Машинное обучение с подкреплением : Учебник / П.В. Никитин, С.А. Корчагин, Е.В. Романова Электрон. дан. Москва : КноРус, 2026 236 с. Режим доступа: book.ru Internet access <https://book.ru/book/959175> ISBN 978-5-406-15268-3. [БИК ID: RU\bookru\bibl\959175]

Нормативные правовые акты:

-

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Электронная библиотека Финансового университета (ЭБ) <http://elib.fa.ru/> (<http://library.fa.ru/files/elibfa.pdf>)
2. Электронно-библиотечная система BOOK.RU <http://www.book.ru>
3. Электронно-библиотечная система "Университетская библиотека ОНЛАЙН" <http://biblioclub.ru/>
4. Электронно-библиотечная система Znanium <http://www.znanium.com>
5. "Деловая онлайн библиотека" издательства "Альпина Паблишер" <http://lib.alpinadigital.ru/en/library>
6. Электронно-библиотечная система издательства "Лань" <https://e.lanbook.com/>
7. Образовательная платформа "ЮРАЙТ" <https://urait.ru/>
8. Научная электронная библиотека eLibrary.ru <http://elibrary.ru>
9. Информационно-образовательный портал Финуниверситета: <https://org.fa.ru>



10. Методические указания для обучающихся по освоению дисциплины

Основные этапы работы студента по дисциплине **"Основы глубокого обучения"**:

1. Предварительная ориентировка в подлежащем изучению учебном материале по программе.
2. Ознакомление с рекомендованной учебной литературой.
3. Слушание и конспектирование лекций, а также выполнение других видов учебной работы.
4. Планирование самостоятельной работы.
5. Обобщение и систематизация информации, взятой из лекций и прочитанной литературы.
6. Выполнение контрольной работы.

Рекомендации по работе с учебным материалом:

1. Осознавайте наличный уровень полученных вами знаний.
2. В ситуации непонимания нужно выявить тот первичный уровень и факторы непонимания, которые стали препятствием понимания последующего.
3. Задавайте сами себе вопросы и пытайтесь ответить на них.

Рекомендации по работе на лекции и с лекционным материалом:

1. Основная задача на лекции – осмысление излагаемого в ней материала. Для этого необходимо слушать лекцию с самого начала, не упуская общих, ориентирующих в материале рассуждений и установок лектора.
2. Ведение записей на лекции важно и полезно для лучшего осмысливания материала, для сохранения информации, с целью ее дальнейшего использования.
3. Для облегчения записи рекомендуется применять сокращения повторяющихся терминов или хорошо известных понятий.

Рекомендации по работе с литературой:

1. Если возникли затруднения при разыскании материала, по какому-либо конкретному вопросу, следует обратиться к предметному указателю, напечатанному, как правило, в конце каждого литературного источника.
2. Предметный указатель – это алфавитный список основных научных понятий (терминов), содержание которых раскрыто в книге, рядом с термином стоят числа, обозначающие номера страниц, на которых изложен материал, относящийся к данному понятию.



Рекомендации по выполнению контрольной работы:

1. Перед выполнением контрольной работы студент должен изучить соответствующие разделы учебной литературы.
2. Контрольную работу студент должен выполнять самостоятельно, используя те навыки и умения, которые получил на лекциях и практических занятиях.
3. При затруднениях, возникших при выполнении контрольной работы, студент может получить консультацию преподавателя

При подготовке к контрольной работе следует использовать нормативные документы Финансового университета, Методические рекомендации по планированию и организации внеаудиторной самостоятельной работы студентов по образовательным программам бакалавриата и магистратуры в Финансовом университете, утвержденные приказом Финуниверситета от 11.05.2021 г. № 1040/о (см. сайт Финансового Университета: на главной странице раздел "Наш университет" → "Единая правовая база Финуниверситета" → "Организация учебного процесса" → "Нормативные документы по самостоятельной работе").



11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

Комплект лицензионного программного обеспечения:

1. Python 3.8
2. Jupyter Notebook (Windows 10)
3. Docker
4. Astra Linux
5. Libre Office
6. Anaconda
7. библиотеки Keras
8. PyTorch
9. Редактор кода VS CODE
10. Антивирус Kaspersky

Современные профессиональные базы данных и информационные справочные системы

1. Электронная энциклопедия: <http://ru.wikipedia.org/wiki/Wiki>
2. Язык программирования Python 3. <https://pythonworld.ru/>
3. Размещение ЭУК: <https://www.campus.fa.ru/>

Сертифицированные программные и аппаратные средства защиты информации

1. не предусмотрены

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

1. Учебная аудитория для проведения учебных занятий, предусмотренных программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная оборудованием и техническими средствами обучения: мебель аудиторная (столы, стулья, доска аудиторная), набор демонстрационного оборудования (проектор, экран)