

Общество с ограниченной ответственностью «1Т»

УТВЕРЖДАЮ

Генеральный директор ООО «1Т»



(В. В. Кармаза)

«08» апреля 2024 г.

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА ПОВЫШЕНИЯ
КВАЛИФИКАЦИИ
«Архитектор данных»

Специальность: Архитектор данных (Data Architect)

Целевое назначение: Использование и разработка технологий искусственного
интеллекта

Срок обучения: 260 академических часов

Москва 2024

ОГЛАВЛЕНИЕ

ОБЩИЕ ДАННЫЕ О ДОПОЛНИТЕЛЬНОЙ ПРОФЕССИОНАЛЬНОЙ ПРОГРАММЕ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ «АРХИТЕКТОР ДАННЫХ»	3
1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ	8
1.1 Актуальность.....	8
1.2 Категория слушателей. Требования к уровню подготовки слушателя	10
1.3. Область профессиональной деятельности	12
1.4 Планируемые результаты обучения.....	12
2. УЧЕБНЫЙ (ТЕМАТИЧЕСКИЙ) ПЛАН.....	20
3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК.....	25
4. РАБОЧИЕ ПРОГРАММЫ МОДУЛЕЙ УЧЕБНОГО КУРСА ПРОГРАММЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ «АРХИТЕКТОР ДАННЫХ»	26
4.1 Рабочая программа модуля 1. Базовый.....	26
4.2 Рабочая программа модуля 2. Профильный.....	46
5. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ	68
6. СИСТЕМА ОЦЕНКИ КАЧЕСТВА ОСВОЕНИЯ ПРОГРАММЫ.....	71

**ОБЩИЕ ДАННЫЕ О ДОПОЛНИТЕЛЬНОЙ ПРОФЕССИОНАЛЬНОЙ ПРОГРАММЕ
ПОВЫШЕНИЯ КВАЛИФИКАЦИИ «АРХИТЕКТОР ДАННЫХ»**

№	Название	Описание
1.1	Название программы	Архитектор данных
1.2	Цель обучения	Получение слушателями компетенций, необходимых для профессиональной деятельности архитектора данных для разработки и применения технологических решений в области искусственного интеллекта и в смежных областях
1.3	Специальность	Архитектор данных (Data Architect)
1.4	Форма обучения	Очно-заочная форма обучения, осуществляемая с применением электронного обучения и дистанционных образовательных технологий (онлайн-вебинары и другие форматы онлайн-обучения) без отрыва от производства
1.5	Количество академических часов	260
1.6	Количество слушателей, которое может обеспечить обучением провайдер по Образовательной программе по одному потоку в срок до 25 ноября	210
1.7.	Стоимость обучения	80 000

Дополнительная профессиональная программа (программа повышения квалификации) «Архитектор данных» (далее – Программа) разработана:

- в соответствии с нормами Федерального закона РФ от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации»;

- с учетом постановления Правительства Российской Федерации от 13 мая 2021 г. № 729 «О мерах по реализации программы стратегического лидерства «Приоритет-2030» (в редакции постановления Правительства Российской Федерации от 14 марта 2022 г. № 357 «О внесении изменений в постановление Правительства Российской Федерации от 13 мая 2021 г. № 729»);

- с учетом требований приказа Минобрнауки России от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам», с изменениями, внесенными приказом

Минобрнауки России от 15 ноября 2013 г. № 1244 «О внесении изменений в Порядок организации и осуществления образовательной деятельности по дополнительным профессиональным программам, утвержденный приказом Министерства образования и науки Российской Федерации от 1 июля 2013 г. № 499»;

– с учетом приказа Министерства образования и науки РФ от 23 августа 2017 г. № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»; паспорта федерального проекта «Искусственный интеллект» национальной программы «Цифровая экономика Российской Федерации»;

– с учетом Методических рекомендаций-разъяснений Минобрнауки России по разработке дополнительных профессиональных программ на основе профессиональных стандартов от 22 апреля 2015 года № ВК-1030/06;

– с учетом приказа Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 28 февраля 2022 г. № 143 «Об утверждении методик расчета показателей федеральных проектов национальной программы «Цифровая экономика Российской Федерации» и признании утратившими силу некоторых приказов Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации об утверждении методик расчета показателей федеральных проектов национальной программы «Цифровая экономика Российской Федерации»;

– с учетом профессионального стандарта 06.042 Специалист по большим данным (приказ Министерства труда и социальной защиты Российской Федерации от 6 июля 2020 г. № 405н), федеральных государственных образовательных стандартов (далее вместе – ФГОС ВО) по направлению подготовки 38.03.05 Бизнес-информатика (приказ Минобрнауки России от 29.07.2020), 09.04.04. Программная инженерия (Приказ Минобрнауки России от 19.09.2017 № 932 (ред. от 08.02.2021));

– на основе анализа требований рынка труда в сфере искусственного интеллекта и анализа данных.

В результате освоения данной программы выпускник программы «Архитектор данных» должен:

Знать:

- функции, задачи, навыки, содержание работы архитектора данных;
- определения, историю развития и главные тренды больших данных и искусственного интеллекта;

- архитектуру, принципы, процесс, стадии и методологии разработки решений на основе ИИ;
- синтаксис и семантику основных языков искусственного интеллекта (NLP, UML, Python, JAVA/C#/C++/Scala) и основные приемы программирования на них;
- методы и инструментальные средства решения задач искусственного интеллекта;
- методы и программный инструментарий для тестирования программных компонентов;
- основные типы, источники больших данных, проблемы безопасности;
- методы и программный инструментарий технологий больших данных;
- базовые понятия и возможности теории игр для их применения в области машинного обучения и искусственного интеллекта;
- системы обработки и анализа больших массивов данных (SQL, NoSQL, Hadoop, SPARK и др.): принципы работы, функционал, язык запросов, преимущества и недостатки;
- технологии, методы и инструменты машинного обучения для решения профессиональных задач;
- нейронные сети: полносвязные, свёрточные и рекуррентные нейронные сети;
- методы обучения нейронных сетей;
- нейросетевые методы понижения размерности;
- архитектуры современных нейронных сетей и их использование для решения профессиональных задач;
- методы, инструментальные средства и стандарты статистического и описательного анализа данных;
- основные платформы данных (облачные и внутрикорпоративные), применяемые при разработке решений на основе ИИ;
- основные виды представления данных: табличные, графовые, временные ряды;
- основные уровни представления данных;
- понятие сквозных цифровых субтехнологий искусственного интеллекта;
- методы и технологии внедрения сквозных цифровых субтехнологий искусственного интеллекта.

Уметь:

- осуществлять поиск информации о новых и перспективных методах анализа больших данных, выполнять сравнительный анализ методов;

- применять методы машинного обучения для тестирования программных компонентов;
- проводить тестирование систем искусственного интеллекта;
- применять методы машинного обучения для работы с большими данными;
- применять методы машинного обучения и нейронных сетей;
- поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов;
- использовать основные статистические методы анализа данных;
- использовать шины данных (kafka) при обработке больших данных;
- моделировать данные в хранилищах (DWH, Data Lake, NoSQL хранилища данных);
- внедрять технологии компьютерного зрения для обработки изображений и видео, задач биометрии;
- внедрять технологии обработки естественного языка для анализа и генерации текстов;
- внедрять технологии обучения с подкреплением для задач автоматизации производства и транспорта.

Владеть:

- языками программирования (Python, JAVA/C#/C++/Scala) для реализации методов анализа и структурирования данных;
- основными инструментами для работы с данными (Git, Docker, CI/CD, Jupyter notebook, kaggle, Nvidia Cuda, VS, IntelliJIdea);
- навыками работы с основными библиотеками Python для визуализации данных (Matplotlib, Seaborn, Plotly);
- навыком создания логической и физической базы данных;
- навыком установки Nadoop на виртуальную машину;
- навыком использования Docker;
- навыком работы с MapReduce, PySpark;
- навыком BI-аналитики;
- навыком работы с NumPy;
- культурой постановки и планирования последовательности решения задач анализа данных и классификации;
- навыками анализа реальных задач из различных предметных областей на уровне отдельных подходов и коллективами алгоритмов;

- навыками работы с библиотеками, программными платформами для автоматизации процессов тестирования;
- навыками работы с библиотеками, программными платформами (фреймворками) и программными комплексами машинного обучения;
- навыком построения полносвязной нейронной сети для задачи классификации;
- основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных;
- способами улучшить качество модели с помощью методов Feature engineering;
- навыком использования библиотеки PyTorch, применения GPU в PyTorch;
- навыком построения полносвязной нейронной сети для задачи классификации;
- основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных;
- методами машинного обучения и нейронных сетей;
- навыком создания логической и физической базы данных;
- навыком BI-аналитики;
- языками запросов для обращения к СУБД;
- навыком работы с генеративными моделями GAN для задач генерации;
- навыком работы с автокодировщиками для задач уменьшения размерности и поиска аномалий;
- навыком работы с технологией RL - обучения с подкреплением.

Разработчики программы:

Борисов Вадим Владимирович, профессор кафедры вычислительной техники, филиал НИУ «МЭИ» в г. Смоленске, д.т.н., профессор,

Мещеряков Роман Валерьевич, главный научный сотрудник ИПУ РАН, д.т.н., профессор,

Панкратов Денис Юрьевич, доцент кафедры «Системы и сети радиосвязи и телерадиовещания» факультета «Радио и телевидение» МТУСИ, к.т.н.,

Набока Михаил Викторович, ведущий BI Data Analyst, индивидуальный предприниматель, к.т.н.,

Борисов Иван Сергеевич, системный аналитик ООО «1Т»,

Запорожец Марина Владимировна, инженер данных ООО «1Т»,

Никаноров Иван Михайлович, ведущий разработчик ООО «1Т»,

Семененко Анатолий Сергеевич, старший преподаватель по ИИ ООО «1Т»

Семчук Дмитрий Борисович, ведущий системный аналитик ООО «1Т»,

Шумаков Максим Витальевич, ведущий разработчик ООО «1Т»,

Абанкина Ирина Всеволодовна, профессор, главный научный сотрудник Центра финансово-экономических решений в образовании, НИУ «Высшая школа экономики», к.э.н.

Трефилов Павел Александрович, разработчик ООО «1Т».

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ.

1.1. Актуальность.

Роль и характер работ в разных сферах экономики Российской Федерации продолжают претерпевать кардинальные реформации – появление новых компетенций и навыков применения инновационных инструментов стимулирует системность и последовательность в вопросах обеспечения непрерывного развития кадров в соответствии с глобальными трендами цифровизации, что предполагает широкое внедрение технологий искусственного интеллекта, анализа больших данных и машинного обучения, которые призваны стать элементом массового образования.

Одним из механизмов, направленных на формирование и развитие оптимальной модели управления системой образования Российской Федерации как весомым общественным институтом, для достижения высоких качественных показателей, повышения социальной и экономической эффективности функционирования национальной системы образования, внедрения инновационных разработок по прорывным технологиям выступает реализация комплекса мероприятий государственной политики в форме государственных программ. В частности, на сегодняшний день в рамках государственной программы «Научно-технологическое развитие Российской Федерации» реализуется множество мероприятий, посвященных вопросам профессиональной подготовки высококвалифицированных кадров для перспективных отраслей экономики Российской Федерации, что обеспечит рост кадрового потенциала цифровой экономики, при этом граждане получают новые возможности и мотивацию для освоения цифровых компетенций.

Согласно «Национальной стратегии развития искусственного интеллекта на период до 2030 года», утверждённой Указом Президента РФ от 10.10.2019 г. № 409 (далее – Стратегия), под искусственным интеллектом в Программе развития понимается комплекс технологических решений, позволяющий имитировать когнитивные функции человека (включая самообучение и поиск решений без заранее заданного алгоритма) и получать при выполнении конкретных задач результаты, сопоставимые, как минимум, с результатами интеллектуальной деятельности человека. Следовательно, необходимо повышение эффективности процессов планирования, прогнозирования и принятия управленческих решений, использование автономного интеллектуального оборудования и робототехнических комплексов, интеллектуальных систем управления логистикой для продуктивной реализации задач указанной Стратегии на основе технологий искусственного интеллекта.

Согласно Стратегии научно-технологического развития Российской Федерации, утвержденной Указом Президента Российской Федерации от 1 декабря 2016 г. № 642, одним из приоритетных направлений научно-технического развития Российской Федерации выступает создание технологий, являющихся основой инновационного развития внутреннего рынка продуктов и услуг, что может быть осуществлено за счет перехода к цифровым, интеллектуальным производственным технологиям, роботизированным системам, новым материалам и способам конструирования, создания систем обработки больших объемов данных, машинного обучения и искусственного интеллекта.

В 2020 году в Российской Федерации был дан старт федеральному проекту «Искусственный интеллект» национального проекта «Цифровая экономика», ключевым направлением которого стало развитие рынка ИИ, а также создания условий использования предприятиями и гражданами продуктов и услуг, основанных на преимущественно отечественных технологиях искусственного интеллекта, посредством которых достигается качественно новый уровень эффективности деятельности.

В условиях постоянного роста потока разнородных данных при реализации различных проектов в области ИИ необходима автоматизация бизнес-процессов, которая должна быть основана на применении действительно нужных данных, что требует повышенного внимания к архитектуре данных проекта, архитектуре управления потоками данных, правилам их обработки. Это значительно повышает роль такого специалиста как архитектор данных. Архитектура данных в современных информационных и аналитических системах должна учитывать и требования конечной задачи, и особенности представления данных для их хранения и обработки, а также специфику и требования трудоёмких алгоритмов анализа данных с использованием технологий ИИ.

Архитектор данных стремится обеспечить организацию, доступность, безопасность и актуальность данных, отвечает за проектирование систем и процессов, которые хранят и используют данные, роль которых чрезвычайно важна. В деятельности архитектора данных определяющим является обеспечение того, чтобы различные элементы системы гармонично сочетались друг с другом, создавая по базе технологий ИИ решение, которое удовлетворяет различные потребности всех конечных пользователей. Таким образом, актуализируется задача подготовки кадров соответствующей квалификации (архитектор данных) с учетом потребностей бизнеса к квалификации специалистов в области искусственного интеллекта.

К ключевым задачам архитектора данных относятся: проектирование логических и физических структур систем данных (хранилища данных), создание моделей данных, определяющих порядок организации, хранения, доступа и обслуживания данных, умение

работать с новыми технологиями и интегрировать их в общую архитектуру данных (облачные вычисления, озера данных), работа с инженерами по данным для обеспечения развития и бесперебойной работы конвейеров данных, процессов ETL и других процессов, связанных с данными, участие в разработке бизнес-стратегии высокого уровня и перевод практических целей организации в практические задачи и проектирование систем.

Образовательная программа в рамках специальности «Архитектор данных», направленная на формирование целостного инженерного видения и практики реализации и применения систем искусственного интеллекта в реальных условиях, с акцентом на технологии программной инженерии обеспечит приобретение следующих профессиональных компетенций:

- способность классифицировать и идентифицировать задачи искусственного интеллекта, выбирать адекватные методы и инструментальные средства решения задач искусственного интеллекта;
- способность применять методы и программные средства автоматизированного логического вывода и автоматизированной проверки гипотез;
- способность разрабатывать и тестировать программные компоненты решения задач в системах искусственного интеллекта;
- способность разрабатывать и применять методы машинного обучения для решения задач;
- способность использовать системы искусственного интеллекта на основе нейросетевых моделей и методов;
- способность осуществлять сбор и подготовку данных для систем искусственного интеллекта;
- способность создавать и поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов;
- способность создавать и внедрять одну или несколько сквозных цифровых субтехнологий искусственного интеллекта.

1.2. Категория слушателей. Требования к уровню подготовки слушателя.

К обучению на программе допускаются: предприниматели, работники и владельцы компаний ИТ-сектора, имеющие высшее образование или среднее профессиональное (либо получающие высшее или среднее профессиональное образование), а также мотивированные специалисты из других профессиональных сфер и студенты, обучающиеся в области

информационных технологий, а также по иным специальностям, которые заинтересованы в получении новых компетенций по специальности «Архитектор данных».

Наличие опыта профессиональной деятельности: без опыта.

Требования к уровню подготовленности, определяемому контрольно-измерительными материалами.

Слушатели должны обладать следующими знаниями, умениями и владеть навыками:

Умение писать программы на каком-либо императивном языке программирования (Java, Python, C# и т.п.).

PYTHON:

Знание синтаксиса языка.

Понимание базовых структур данных.

Иметь базовое представление о работе популярных алгоритмах (бинарный поиск, рекурсия и т. д.).

Базовое представление о скорости работы алгоритмов («O» большое).

Иметь базовое представление о работе с модулем Pandas и NumPy.

SQL:

Знание базового синтаксиса (SELECT, WHERE, GROUP BY, HAVING).

Умение составлять подзапросы и делать все виды JOIN.

Навык работы с вложенными запросами.

ИНФРАСТРУКТУРА:

Умение работать с командной строкой.

Знание базовых команд Linux.

Навыки работы с Git.

Базовое знание Docker.

МАШИННОЕ ОБУЧЕНИЕ:

Знание классических моделей (линейные модели и решающие деревья).

Знание классических метрик качества модели (accuracy, MSE, RMSE).

Требования к компетенциям, которыми должен обладать гражданин при поступлении на программу:

– способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач;

– способен применять и модифицировать математические модели для решения задач в области профессиональной деятельности;

– владеет широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий;

– способен использовать современные компьютерные технологии поиска информации для решения поставленной задачи, критического анализа этой информации и обоснования принятых идей и подходов к решению;

– способен выбирать и оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно-) для решения поставленной задачи.

1.3. Область профессиональной деятельности.

06 Связь и информационно-коммуникационные технологии (в сферах: анализа, моделирования и формирования интегрального представления стратегий и целей, бизнес-процессов и информационно-технологической инфраструктуры предприятий различной отраслевой принадлежности и различных форм собственности, а также учреждений государственного и муниципального управления; стратегического планирования и управления развитием информационных систем и информационно-коммуникационных технологий управления предприятием; аналитической поддержки процессов принятия решений для управления предприятием).

1.4. Планируемые результаты обучения.

Программа повышения квалификации разработана с учетом профессиональных стандартов:

«Специалист по большим данным», утвержденный приказом Министерства труда и социальной защиты РФ от 6 июля 2020 года №405н;

«Программист», утвержденный приказом Министерства труда и социальной защиты РФ от 20 июля 2022 № 424н.

Программа повышения квалификации разработана с учетом:

ФГОС 38.03.05 Бизнес-информатика, ФГОС 09.04.04. Программная инженерия.

По данной программе приобретаются компетенции универсальной модели компетенций в сфере искусственного интеллекта, разработанной РЭУ им. Г.В. Плеханова в 2021 году в рамках результата Федерального проекта «Искусственный интеллект».

Совершенствуемые и/или формируемые компетенции	Тип компетенции	Планируемые результаты обучения (знать, уметь, владеть — использовать конкретные инструменты)
Способен классифицировать и идентифицировать задачи искусственного интеллекта, выбирать адекватные методы и инструментальные средства решения задач искусственного интеллекта	профессиональная	<p>Знать:</p> <ul style="list-style-type: none"> – определения, историю развития и главные тренды искусственного интеллекта; – архитектуру, принципы, процесс, стадии и методологии разработки решений на основе ИИ; – методы и инструментальные средства решения задач искусственного интеллекта. <p>Уметь:</p> <ul style="list-style-type: none"> – осуществлять поиск информации о новых и перспективных методах анализа больших данных, выполнять сравнительный анализ методов. <p>Владеть:</p> <ul style="list-style-type: none"> – культурой постановки и планирования последовательности решения задач анализа данных и классификации; – навыками анализа реальных задач из различных предметных областей на уровне отдельных подходов и коллективами алгоритмов.
Способен применять методы и программные средства автоматизированного логического вывода и автоматизированной проверки гипотез	профессиональная	<p>Знать:</p> <ul style="list-style-type: none"> – методы автоматизированной генерации и проверки гипотез в сфере исследовательской деятельности; – основные принципы логического вывода и проверки гипотез в контексте анализа данных; – теоретические основы алгоритмов, используемых для автоматизированного логического вывода и проверки гипотез; – инструменты и технологии, применяемые для реализации

		<p>автоматизированного логического вывода и проверки гипотез в сфере анализа данных.</p> <p>Уметь:</p> <ul style="list-style-type: none"> – настраивать, конфигурировать и адаптировать программные средства автоматизированного логического вывода в сфере исследовательской деятельности; – разрабатывать и настраивать алгоритмы автоматизированного логического вывода и проверки гипотез, учитывая специфику исследовательской деятельности; – проектировать и настраивать конфигурации программных средств для эффективного применения методов автоматизированного логического вывода и проверки гипотез; – проводить адаптацию и оптимизацию программных решений для учета требований и специфики конкретных задач. <p>Владеть:</p> <ul style="list-style-type: none"> – программными инструментами, позволяющими осуществлять автоматизированный логический вывод и проверку гипотез, такими как инструменты теории машинного обучения, статистические пакеты, инструменты символьного вычисления и другие соответствующие технологии; – техниками и методами, используемыми для применения программных средств автоматизированного логического вывода в различных областях исследовательской деятельности; – навыками использования различных инструментов и технологий в рамках процесса автоматизации логического вывода и проверки гипотез
Способен разрабатывать и тестировать программные компоненты решения задач	профессиональная	<p>Знать:</p> <ul style="list-style-type: none"> – функции, задачи, навыки, содержание работы архитектора

<p>в системах искусственного интеллекта</p>		<p>данных;</p> <ul style="list-style-type: none"> – архитектуру, принципы, процесс, стадии и методологии разработки решений на основе ИИ; – синтаксис и семантику основных языков искусственного интеллекта (Python, JAVA/C#/C++/Scala) и основные приемы программирования на них; – среды разработки программных компонентов; – решение задач в системах искусственного интеллекта; – методы и программный инструментарий для тестирования программных компонентов. <p>Уметь:</p> <ul style="list-style-type: none"> – применять методы машинного обучения для работы с большими данными; – применять методы машинного обучения для тестирования программных компонентов; – проводить тестирование систем искусственного интеллекта. <p>Владеть:</p> <ul style="list-style-type: none"> – языками программирования (Python, JAVA/C#/C++/Scala) для разработки программных компонентов решения задач в системах ИИ; – основными инструментами для работы с данными (Git, Docker, CI/CD, Jupyter notebook, kaggle, Nvidia Cuda, VS, IntelliJIdea); – навыками работы с основными библиотеками Python (Matplotlib, Seaborn, Plotly); – навыками работы с библиотеками, программными платформами для автоматизации процессов тестирования.
<p>Способен разрабатывать и применять методы машинного обучения для решения задач</p>	<p>профессиональная</p>	<p>Знать:</p> <ul style="list-style-type: none"> – основные типы, источники больших данных, проблемы безопасности; – методы и программный инструментарий технологий больших данных; – базовые понятия и возможности теории игр для их применения в

		<p>области машинного обучения и искусственного интеллекта;</p> <ul style="list-style-type: none"> – системы обработки и анализа больших массивов данных (SQL, NoSQL, Hadoop, SPARK и др.): принципы работы, функционал, язык запросов, преимущества и недостатки; – технологии, методы и инструменты машинного обучения для решения профессиональных задач. <p>Уметь:</p> <ul style="list-style-type: none"> – применять методы машинного обучения для работы с большими данными. <p>Владеть:</p> <ul style="list-style-type: none"> – навыками работы с библиотеками, программными платформами (фреймворками) и программными комплексами машинного обучения
Способен использовать системы искусственного интеллекта на основе нейросетевых моделей и методов	профессиональная	<p>Знать:</p> <ul style="list-style-type: none"> – нейронные сети: полносвязные, свёрточные и рекуррентные нейронные сети; – методы обучения нейронных сетей; – нейросетевые методы понижения размерности. <p>Уметь:</p> <ul style="list-style-type: none"> – применять методы машинного обучения и нейронных сетей. <p>Владеть:</p> <ul style="list-style-type: none"> – навыком построения полносвязной нейронной сети для задачи классификации; – основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных.
Способен создавать и поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов	профессиональная	<p>Знать:</p> <ul style="list-style-type: none"> – архитектуры современных нейронных сетей и их использование для решения профессиональных задач <p>Уметь:</p> <ul style="list-style-type: none"> – поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов <p>Владеть:</p> <ul style="list-style-type: none"> – способами улучшить качество модели с помощью методов Feature

		<p>engineering;</p> <ul style="list-style-type: none"> – навыком использования библиотеки PyTorch, применения GPU в PyTorch; – навыком построения полносвязной нейронной сети для задачи классификации; – основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных; – методами машинного обучения и нейронных сетей.
Способен осуществлять сбор и подготовку данных для систем искусственного интеллекта	профессиональная	<p>Знать:</p> <ul style="list-style-type: none"> – методы, инструментальные средства и стандарты статистического и описательного анализа данных; – основные платформы данных (облачные и внутрикорпоративные), применяемые при разработке решений на основе ИИ; – основные виды представления данных: табличные, графовые, временные ряды; – основные уровни представления данных; <p>Уметь:</p> <ul style="list-style-type: none"> – использовать основные статистические методы анализа данных; – моделировать данные в хранилищах (DWH, Data Lake, NoSQL хранилища данных); <p>Владеть:</p> <ul style="list-style-type: none"> – навыком создания логической и физической базы данных; – навыком BI-аналитики; – языками запросов для обращения к СУБД.
Способен создавать и внедрять одну или несколько сквозных цифровых субтехнологий искусственного интеллекта	профессиональная	<p>Знать:</p> <ul style="list-style-type: none"> – понятие сквозных цифровых субтехнологий искусственного интеллекта; – методы и технологии внедрения сквозных цифровых субтехнологий искусственного интеллекта. <p>Уметь:</p> <ul style="list-style-type: none"> – внедрять технологии компьютерного зрения для обработки

		<p>изображений и видео, задач биометрии;</p> <ul style="list-style-type: none"> – внедрять технологии обработки естественного языка для анализа и генерации текстов; – внедрять технологии обучения с подкреплением для задач автоматизации производства и транспорта. <p>Владеть:</p> <ul style="list-style-type: none"> – навыком работы с генеративными моделями GAN для задач генерации; – навыком работы с автокодировщиками для задач уменьшения размерности и поиска аномалий; – навыком работы с технологией RL - обучения с подкреплением.
--	--	---

Критерии для оценки уровня сформированности указанных компетенций и соответствия индикаторам достижения компетенций

Компетенция	Критерии для оценки уровня сформированности компетенций
Способен классифицировать и идентифицировать задачи искусственного интеллекта, выбирать адекватные методы и инструментальные средства решения задач искусственного интеллекта	Классифицирует и идентифицирует задачи систем искусственного интеллекта в зависимости от особенностей проблемной и предметной областей. Выбирает методы и инструментальные средства искусственного интеллекта для решения задач в зависимости от особенностей проблемной области. Собирает исходную информацию и формирует требования к решению задач с использованием методов искусственного интеллекта
Способен применять методы и программные средства автоматизированного логического вывода и автоматизированной проверки гипотез	Использует методы автоматизированной генерации и проверки гипотез в сфере исследовательской деятельности. Настраивает, конфигурирует и адаптирует программные средства автоматизированного логического вывода в сфере исследовательской деятельности
Способен разрабатывать и тестировать программные компоненты решения задач в системах искусственного интеллекта	Проводит тестирование систем искусственного интеллекта
Способен разрабатывать и применять методы машинного обучения для решения задач	Проводит анализ требований и определяет необходимые классы задач машинного обучения. Принимает участие в оценке и выборе используемых методов машинного обучения
Способен использовать системы искусственного интеллекта на основе нейросетевых моделей и методов	Осуществляет оценку и выбор моделей искусственных нейронных сетей и инструментальных средств для решения

	поставленной задачи
Способен осуществлять сбор и подготовку данных для систем искусственного интеллекта	Выполняет подготовку и разметку структурированных и неструктурированных данных для машинного обучения
Способен создавать и поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов	Осуществляет оценку и выбор моделей искусственных нейронных сетей и инструментальных средств для решения поставленной задачи
Способен осуществлять сбор и подготовку данных для систем искусственного интеллекта	Осуществляет поиск данных в открытых источниках, специализированных библиотеках и репозиториях
Способен создавать и внедрять одну или несколько сквозных цифровых субтехнологий искусственного интеллекта	Участствует в реализации проектов в области сквозной цифровой субтехнологии «Компьютерное зрение»; Участствует в реализации проектов в области сквозной цифровой субтехнологии «Обработка естественного языка»

2. УЧЕБНЫЙ (ТЕМАТИЧЕСКИЙ) ПЛАН.

№ п/п	Наименование модулей/тем программы	Всего, час	Виды учебных занятий			Формы контроля
			лекции	практические занятия	самостоятельная работа	
Модуль 1. Базовый						
Раздел 1.1 Введение в архитектуру больших данных						
1	Тема 1. Введение в профессию. Основные принципы работы с большими данными	6	2	2	2	тест, практическая работа
2	Тема 2. Технологии анализа данных. Статистический и описательный анализ данных	8	2	4	2	тест, практическая работа
3	Тема 3. Введение в основные инструменты для работы с данными. Основы GIT на примере GitLab/GitHub	8	2	4	2	тест, практическая работа
4	Тема 4. Введение в контейнеризацию с Docker. Управление контейнерами и создание образов, Docker-compose	12	2	6	4	тест, практическая работа
5	Тема 5. Введение в языки программирования. Основные популярные языки для ИИ. Введение в Python	10	2	6	2	тест, практическая работа
6	Тема 6. Работа с основными библиотеками Python для обработки, вычислений и	10	2	6	2	тест, практическая работа

	визуализации данных (Pandas, NumPy, SciPy, Matplotlib, Seaborn, Plotly)					
Раздел 1.2 Введение в системное администрирование больших данных						
7	Тема 1. Знакомство с Linux и его история. Различные дистрибутивы Linux и выбор подходящего для конкретной задачи. Установка Linux	8	2	4	2	тест, практическая работа
8	Тема 2. Основа работы с командной строкой Linux. Управление пользователями, группами и правами доступа	12	2	6	4	тест, практическая работа
9	Тема 3. Введение в пакетные менеджеры и установка пакетов обработки больших данных. Работа со сторонними репозиториями	6	1	3	2	тест, практическая работа
10	Тема 4. Управление процессами и службами обработки больших данных. Мониторинг системы и журналы	12	2	6	4	тест, практическая работа
11	Тема 5. Основы безопасности больших данных. Настройка сети и брандмауэры	6	1	3	2	тест, практическая работа
12	Тема 6. Создание и редактирование скриптов на Bash для обработки больших данных. Использование утилиты Cron для	8	2	4	2	тест, практическая работа

	планирования задач					
	Промежуточная аттестация	4		4		Решение кейса
Модуль 2. Профильный						
Раздел 2.1 Проектирование архитектур: большие данные, машинное обучение, искусственный интеллект						
13	Тема 1. Введение в базы данных и теорию хранения данных. Реляционные базы данных, использование SQL	8	2	4	2	тест, практическая работа
14	Тема 2. Введение в Hadoop. DWH, Data Lake, NoSQL хранилища данных	8	2	4	2	тест, практическая работа
15	Тема 3. Массово-параллельная обработка и анализ данных Работа с PySpark. Поточковая обработка данных (data streaming, event processing). Шины данных (kafka)	8	2	4	2	тест, практическая работа
16	Тема 4. Введение в ML. Основные задачи, виды классических моделей, метрики качества моделей. BI-системы и визуализация данных	8	2	4	2	тест, практическая работа
17	Тема 5. Погружение в ML. Обучение с учителем	10	2	6	2	тест, практическая работа
18	Тема 6. Погружение в ML. Обучение без учителя	10	2	6	2	тест, практическая работа
19	Тема 7. Введение в нейронные сети.	8	2	4	2	тест, практическая

	Процесс, стадии и методологии разработки решений на основе искусственного интеллекта					работа
20	Тема 8. CV - компьютерное зрение	10	2	4	4	тест, практическая работа
21	Тема 9. NLP – обработка естественного языка	10	2	4	4	тест, практическая работа
Раздел 2.2. Построение архитектур больших данных						
22	Тема 1. Введение в Kubernetes и установка Minikube. Управление ресурсами архитектур больших данных в Kubernetes	12	2	6	4	тест, практическая работа
23	Тема 2. Масштабирование и автомасштабирование архитектур больших данных. Мониторинг в Kubernetes	12	2	6	4	тест, практическая работа
24	Тема 3. Управление пакетами обработки больших данных с Helm	6	1	3	2	тест, практическая работа
25	Тема 4. CI/CD пакетов обработки больших данных в Kubernetes. Основы CI/CD на примере Gitlab CI	16	2	6	8	тест, практическая работа
26	Тема 5. DataOps в Kubernetes с Apache Airflow	6	1	3	2	тест, практическая работа
27	Тема 6. MLOps в Kubernetes с Kubeflow	6	1	3	2	тест, практическая работа

	Промежуточная аттестация	4		4		Решение кейса
	Итоговая аттестация	8		8		Решение кейсов
	Всего часов	260	49	137	74	–

3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК.

Объем программы – 260 часов.

Продолжительность обучения – 3 месяца.

Форма обучения – очно-заочная с применением электронного обучения и дистанционных образовательных технологий.

Режим занятий: 3-4 часа в день.

Завершение обучения: 25.11.2024

№ п/п	Наименование учебных модулей/ практик/ аттестации	Трудоёмкость (час)	Срок освоения (кол-во учебных дней)
1	Модуль 1. Базовый. Раздел 1.1 Введение в архитектуру больших данных	54	20
2	Модуль 1. Базовый. Раздел 1.2 Введение в системное администрирование больших данных	52	18
3	Промежуточная аттестация. Решение практико-ориентированной задачи (кейса)	4	1
4	Модуль 2. Профильный. Раздел 2.1 Проектирование архитектур: большие данные, машинное обучение, искусственный интеллект	80	26
5	Модуль 2. Профильный. Раздел 2.2 Построение архитектур больших данных	58	22
6	Промежуточная аттестация. Решение практико-ориентированной задачи (кейса)	4	1
7	Итоговая аттестация	8	2

4. РАБОЧИЕ ПРОГРАММЫ МОДУЛЕЙ УЧЕБНОГО КУРСА ПРОГРАММЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ «АРХИТЕКТОР ДАННЫХ».

Программа повышения квалификации состоит из 2 учебных модулей:

Модуль 1. Базовый.

Раздел 1.1. Введение в архитектуру больших данных.

Раздел 1.2. Введение в системное администрирование больших данных.

Модуль 2. Профильный.

Раздел 2.1. Проектирование архитектур: большие данные, машинное обучение, искусственный интеллект.

Раздел 2.2. Построение архитектур больших данных.

4.1. Рабочая программа модуля 1. Базовый.

Цель освоения модуля 1 – приобретение слушателями знаний по основным принципам работы с большими данными, изучение технологии анализа данных, основного инструментария для работы с данными, изучение технологии системного администрирования больших данных; приобретение практических навыков работы с языками программирования (Python, JAVA/C#/C++/Scala) и другими инструментами для реализации методов анализа и структурирования больших данных; практическое применение методов машинного обучения для работы с большими объемами данных.

Профессиональные компетенции, совершенствуемые и приобретаемые слушателями в процессе освоения модуля 1:

ПК-1.р. Способен классифицировать и идентифицировать задачи искусственного интеллекта, выбирать адекватные методы и инструментальные средства решения задач искусственного интеллекта.

ПК-1.и. Способен применять методы и программные средства автоматизированного логического вывода и автоматизированной проверки гипотез.

ПК-4.р. Способен разрабатывать и применять методы машинного обучения для решения задач.

ПК-5.п. Способен использовать системы искусственного интеллекта на основе нейросетевых моделей и методов.

ПК-6.п. Способен осуществлять сбор и подготовку данных для систем искусственного интеллекта.

ПК-6.р. Способен создавать и поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов.

Планируемые результаты обучения по модулю 1.

По итогам освоения модуля слушатели должны:

Знать:

- определения, историю развития и главные тренды искусственного интеллекта;
- методы автоматизированной генерации и проверки гипотез в сфере исследовательской деятельности;
- основные принципы логического вывода и проверки гипотез в контексте анализа данных;
- теоретические основы алгоритмов, используемых для автоматизированного логического вывода и проверки гипотез;
- архитектуру, принципы, процесс, стадии и методологии разработки решений на основе ИИ;
- методы и инструментальные средства решения задач искусственного интеллекта.
- основные типы, источники больших данных, проблемы безопасности;
- методы и программный инструментарий технологий больших данных;
- базовые понятия и возможности теории игр для их применения в области машинного обучения и искусственного интеллекта;
- системы обработки и анализа больших массивов данных (SQL, NoSQL, Hadoop, SPARK и др.): принципы работы, функционал, язык запросов, преимущества и недостатки;
- технологии, методы и инструменты машинного обучения для решения профессиональных задач;
- нейронные сети: полносвязные, свёрточные и рекуррентные нейронные сети;
- методы обучения нейронных сетей;
- нейросетевые методы понижения размерности;
- методы, инструментальные средства и стандарты статистического и описательного анализа данных;
- основные платформы данных (облачные и внутрикорпоративные), применяемые при разработке решений на основе ИИ;
- основные виды представления данных: табличные, графовые, временные ряды;
- основные уровни представления данных.

Уметь:

- осуществлять поиск информации о новых и перспективных методах анализа больших данных, выполнять сравнительный анализ методов;
- применять методы машинного обучения для работы с большими данными;
- применять методы машинного обучения и нейронных сетей;
- использовать основные статистические методы анализа данных;

Владеть:

- языками программирования (Python, JAVA/C#/C++/Scala) для реализации методов анализа и структурирования данных;
- навыками работы с основными библиотеками Python для визуализации данных (Matplotlib, Seaborn, Plotly);
- культурой постановки и планирования последовательности решения задач анализа данных и классификации;
- навыками анализа реальных задач из различных предметных областей на уровне отдельных подходов и коллективами алгоритмов;
- навыками работы с библиотеками, программными платформами (фреймворками) и программными комплексами машинного обучения;
- навыком построения полносвязной нейронной сети для задачи классификации;
- основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных;
- навыком создания логической и физической базы данных.

Учебно-тематический план модуля 1 Базовый.

№ п/п	Наименование дисциплины, модуля, темы	Трудоемкость		В том числе				Форма контроля
		В зачетных единицах	В часах	Всего	Контактная работа		Самостоятельная работа	
					Лекции	Практические занятия		
1	2	3	4	5	6	7	8	9
Раздел 1.1 Введение в архитектуру больших данных								
1	Тема 1. Введение в профессию. Основные принципы работы с большими данными		6	4	2	2	2	тест, практическая работа
2	Тема 2. Технологии анализа данных. Статистический и описательный анализ		8	6	2	4	2	тест, практическая

	данных							работа
3	Тема 3. Введение в основные инструменты для работы с данными. Основы GIT на примере GitLab/GitHub	8	6	2	4	2		тест, практическая работа
4	Тема 4. Введение в контейнеризацию с Docker. Управление контейнерами и создание образов, Docker-compose	12	8	2	6	4		тест, практическая работа
5	Тема 5. Введение в языки программирования. Основные популярные языки для ИИ. Введение в Python	10	8	2	6	2		тест, практическая работа
6	Тема 6. Работа с основными библиотеками Python для обработки, вычислений и визуализации данных (Pandas, NumPy, SciPy, Matplotlib, Seaborn, Plotly)	10	8	2	6	2		тест, практическая работа
Раздел 1.2 Введение в системное администрирование больших данных								
7	Тема 1. Знакомство с Linux и его история. Различные дистрибутивы Linux и выбор подходящего для конкретной задачи. Установка Linux	8	6	2	4	2		тест, практическая работа
8	Тема 2. Основа работы с командной строкой Linux. Управление пользователями, группами и правами доступа	12	8	2	6	4		тест, практическая работа
9	Тема 3. Введение в пакетные менеджеры и установка пакетов обработки больших данных. Работа со сторонними репозиториями	6	4	1	3	2		тест, практическая работа
10	Тема 4. Управление процессами и службами обработки больших данных. Мониторинг системы и журналы	12	8	2	6	4		тест, практическая работа
11	Тема 5. Основы безопасности больших данных. Настройка сети и брандмауэры	6	4	1	3	2		тест, практическая работа
12	Тема 6. Создание и редактирование скриптов на Bash для обработки больших данных. Использование утилиты Cron для планирования задач	8	6	2	4	2		тест, практическая работа
15	Промежуточная аттестация	4	4		4			Решение кейса
16	Итого по модулю 1	110	80	22	58	30		Решение кейса

Раздел 1.1 Введение в архитектуру больших данных.

Тема 1. Введение в профессию. Основные принципы работы с большими данными.

Архитектор данных: потребность и ценность. Задачи, навыки. Обязанности и функция в команде. Перспективы развития больших данных в мире и в РФ. Этические аспекты

больших данных. Вопросы ответственности и контроля в разработке и использовании больших данных.

Тема 2. Технологии анализа данных. Статистический и описательный анализ данных.

Стандарты жизненного цикла Big Data: CRISP-DM. Когнитивный анализ данных. Data Mining. Методы Data Mining. Предобработка данных (Data Preprocessing). Data quality качество данных, подходы и инструменты Weka, Rapid Miner, Knime, Orange IBM SPSS Modeler, Tableau и др. Введение в математическую статистику и использование ее при анализе данных. Современные методы и инструментальные средства анализа больших данных. Стандарты проведения анализа данных.

Тема 3. Введение в основные инструменты для работы с данными. Основы GIT на примере GitLab/GitHub.

Основные концепции и понятия инструментария работы с большими данными. Как используется в бизнесе. Платформы: Jupiter notebook, kaggle. Среды разработки: VS, IntelliJ Idea. Основы системы GIT. Обзор возможностей сервиса GitHub. Регистрация и работа с сервисом GitHub. Введение в платформу GitLab.

Тема 4. Введение в контейнеризацию с Docker. Управление контейнерами и создание образов, Docker-compose.

Введение в контейнеризацию. Установка Docker и Docker Compose. Разворачивание контейнеров с помощью Docker. Команды Docker. Разворачивание проектов с помощью Docker Compose. Образы Docker. Dockerfile. Слои. Оптимизация размера образов. Мультистейдж сборка. Работа с Docker-реестрами: DockerHub.

Тема 5. Введение в языки программирования. Основные популярные языки для ИИ. Введение в Python.

Существующие и перспективные методы и программный инструментарий технологий больших данных. Введение в Python. Операторы языка. Типы данных и переменные. Базовые конструкции и структуры языка. Циклы и условия. Функции. Сетевые запросы. Словари и множества. Библиотеки. Синтаксис языка программирования JAVA/C#/C++/Scala.

Тема 6. Работа с основными библиотеками Python для обработки, вычислений и визуализации данных (Pandas, NumPy, SciPy, Matplotlib, Seaborn, Plotly).

Работа с табличными данными в Python. Библиотеки Pandas, как стандарт обработки данных. Математический и числовой анализ с помощью библиотек NumPy, SciPy. Использование NumPy, SciPy для научных вычислений. Построение графиков и визуализация данных с помощью инструментов Matplotlib, Seaborn, Plotly.

Раздел 1.2 Введение в системное администрирование больших данных.

Тема 1. Знакомство с Linux и его история. Различные дистрибутивы Linux и выбор подходящего для конкретной задачи. Установка Linux.

История Linux систем. Основы архитектуры Linux. Типы дистрибутивов Linux и их отличия. Установка и настройка VirtualBox. Загрузка и процесс установки Linux на виртуальную машину. Обзор дистрибутива и базовая настройка.

Тема 2. Основа работы с командной строкой Linux. Управление пользователями, группами и правами доступа.

Типы терминалов в Linux системах и их отличия. Базовые утилиты Linux и принципы работы с интерфейсом командной строки. Работа с базовыми файловыми командами Linux. Понятие пользователя и группы. Управление пользователями и группами. Понятие прав доступа в Linux. Настройка прав доступа к файлам и директориям.

Тема 3. Введение в пакетные менеджеры и установка пакетов для обработки больших данных. Работа со сторонними репозиториями.

Пакетные менеджеры и их типы. Репозитории пакетов. Пакетный менеджер APT. Установка, обновление и удаление программ.

Тема 4. Управление процессами и службами обработки больших данных. Мониторинг системы и журналы.

Основы процессов в Linux. Управление процессами. Создание и управление службами. Автозапуск служб. Системы мониторинга и журналирования. Инструменты top и htop. Определение load average и iops. Журналирование. AuditD.

Тема 5. Основы безопасности больших данных. Настройка сети и брандмауэры.

Схемы и методы безопасности Linux. Способы ограничения выполнения в AppArmor и SELinux. Утилиты для мониторинга Sysdig и Falco. Брандмауэры.

Тема 6. Создание и редактирование скриптов на Bash для обработки больших данных. Использование утилиты Cron для планирования задач.

Введение в Bash Scripting. Основы Bash Scripting. Функции в Bash. Работа с файлами и каталогами. Утилита Cron.

Содержание практических занятий по модулю 1.

№ темы	Наименование (содержание) темы, по которой предусмотрено практическое занятие	Формы и методы проведения
Раздел 1.1 Введение в архитектуру больших данных		
1	Провести анализ рынка Архитектор данных в РФ.	Решение практических заданий
2	Провести анализ данных используя	Решение практических

	методологию CRISP-DM. Решить практические задачи используя математическую статистику	заданий
3	Введение в использование Jupyter для чтения и формирования данных на python, как настроить окружение. Выполнить задания по работе с GIT: работа с репозиторием, работа с ветками и их слияние, разрешение конфликтов	Решение практических заданий
4	Установить Docker и Docker Compose, скачать и запустить контейнер Ubuntu, написать docker-compose.yml для запуска нескольких контейнеров. Создать образ Docker и отправить в DockerHub	Решение практических заданий
5	Решить практические задачи используя базовые конструкции и структуры языка. Решить практические задачи на Python, используя основные функции для работы со списками и кортежами, структуру словарей и множеств.	Решение практических заданий
6	Решить практические задачи на Python, используя библиотеки Pandas, NumPy и Scipy. Поиск закономерностей с помощью визуализации данных.	Решение практических заданий
Раздел 1.2 Введение в системное администрирование больших данных		
1	Разобрать 3 представленных кейса. Подобрать дистрибутив под каждый из случаев и дать обоснование своего выбора. Выполнить задания: загрузить и установить VirtualBox, загрузить дистрибутив Linux с официального сайта, создать новую виртуальную машину и запустить процесс установки Linux, авторизоваться в ОС, при невозможности запуска терминала - исправить ошибки	Решение практических заданий
2	Выполнить набор базовых заданий по работе с командной строкой: вывести текущую директорию, создать папку и файл с текстом, вывести подробный листинг текущей директории, добавить еще файл в корневой директории ОС с листингом корневой директории, вывести в консоль содержимое файла и количество строк в файле, написать команду (по индивидуальному заданию), создать новый текстовый файл и вывести последние 10 строк файла в консоль и вывести строки, где встречается определенная подстрока в консоль. Выполнить набор базовых заданий по работе с пользователями и группами: создать пользователя и группу, добавить пользователя в группу и вывести список пользователей	Решение практических заданий

	<p>группы, переключиться на созданного пользователя и вывести текущую директорию, а затем создать в домашней директории пользователя папку с 3 файлами и вывести детальный листинг директории</p>	
3	<p>Выполнить набор базовых заданий по работе с пакетными менеджерами: Обновить список пакетов из репозитория, установить спасибо программы согласно индивидуальных заданий, вывести версии установленных программ через консоль, а также запустить установленные программы</p>	Решение практических заданий
4	<p>Запустить свой процесс, желательно какой-либо скрипт на Python или команду ping, найти PID процесса.</p> <p>Выполнить набор базовых заданий по работе с процессами: создать скрипт, который выполняет некоторую вычислительную задачу (по индивидуальному заданию), запустить этот скрипт в фоновом режиме, найти свой фоновый процесс в списке, остановить свой фоновый процесс, изменить приоритет выполнения своего фонового процесса, убить процесс.</p> <p>Выполнить набор базовых заданий по работе со службами: создать собственную SysV init или systemd службу (по индивидуальному заданию) и подготовить описание службы, включая ее назначение и инструкции по установке и управлению.</p> <p>Выполнить набор базовых заданий по работе с мониторингом: изучить команды top и htop а также iotop и iftop согласно индивидуальных заданий, создать отчета по результату выполнения заданий, описать, какие процессы вызывают наибольшую нагрузку на вашу систему, а также дать рекомендации по оптимизации системы на основе данных, полученных с помощью инструментов мониторинга.</p> <p>Выполнить набор базовых заданий по работе с логированием: просмотреть аудит событий входа в систему и определить количество неудачных попыток входа в систему произошедших в течение последних суток, отфильтровать сообщения по разным критериям, исследовать возможности аудита изменения файлов и журнал генерируемый сервисом</p>	Решение практических заданий
5	<p>Выполнить набор базовых заданий по работе с сетью: включить AppArmor и создать профиль</p>	Решение практических заданий

	<p>для сервиса (по индивидуальному заданию), проверить текущий статус SELinux, настроить публичный DNS, создайте SELinux-политику для сервиса или приложения (по индивидуальному заданию), исследовать дополнительные опции и команды для работы с SELinux и AppArmor, создать подключение со статическим IP-адресом.</p> <p>Выполнить набор базовых заданий по работе с брандмауэром: включить брандмауэр и прописать правило разрешающее подключение для сервиса ssh, загрузить ClamAV, обновить базу, провести проверку, настроить статический IP-адрес, создать правило iptables, разрешающее доступ к порту 22 (SSH) с другого компьютера, исследовать другие параметры конфигурации SSH, такие как настройка доступа по ключам, отключение доступа с помощью пароля, и настройка SSH-туннелей, а затем создать более сложные правила iptables, которые разрешают или блокируют доступ к различным портам или сетевым интерфейсам</p>	
6	<p>Создать запись в стоп при которой каждый час будет посещаться сайт (по индивидуальному заданию) и при помощи команды curl с ключом -o выводиться его содержимое в файл с текущим датой и временем.</p> <p>Создать bash-скрипт, который будет производить ping указанного как параметр сайта (должен спрашивать при запуске dns-name сайта и производить ping). Скрипт должен выгружать все доступные логи в одну папку, архивировать их каждый день и удалять архивы старше 30 дней</p>	Решение практических заданий

Содержание самостоятельной работы слушателей по модулю 1.

Основная цель самостоятельной работы слушателей – закрепление знаний, полученных в ходе лекционных и практических занятий.

Индивидуальная консультационная работа преподавателей со слушателями осуществляется весь период обучения.

№ темы	Наименование (содержание) темы, по которой предусмотрена самостоятельная работа	Формы и методы проведения
Раздел 1.1 Введение в архитектуру больших данных		
1	Тема 1. Введение в профессию. Основные принципы работы с большими данными	Сформулировать задачи архитектора данных на основе проанализированных вакансий.

		Тест: Функции Архитектора данных
2	Тема 2. Технологии анализа данных. Статистический и описательный анализ данных	Изучить материал: Методы анализа на графах. Случайные графы, без масштабные графы, социальные сети – сети тесного мира. Закономерности, методы кластеризации на графах. Рассмотреть дополнительные способы статистического анализа данных. Тест: Технологии анализа данных
3	Тема 3. Введение в основные инструменты для работы с данными. Основы GIT на примере GitLab/GitHub	Изучить дополнительную документацию по Jupyter и Git (GitFlow). Тест: Основные инструменты для работы с данными
4	Тема 4. Введение в контейнеризацию с Docker. Управление контейнерами и создание образов, Docker Compose	Рассмотреть дополнительную официальную документацию (Docker и Docker Compose) по запуску своего ПО, управлению репозиториями и спецификации Dockerfile. Тест: Введение в контейнеризацию
5	Тема 5. Введение в языки программирования. Основные популярные языки для ИИ. Введение в Python	Рассмотреть основные алгоритмы, используемые в языках программирования. Произвести доработку среды разработки Python. Тест: Основы синтаксиса и структур в Python
6	Тема 6. Работа с основными библиотеками Python для обработки, вычислений и визуализации данных (Pandas, NumPy, SciPy, Matplotlib, Seaborn, Plotly)	Определить преимущества и недостатки библиотеки Pandas. Изучить отличия SciPy от NumPy Изучить дополнительные инструменты для построения графиков Тест: Использование библиотек Python
Раздел 1.2 Введение в системное администрирование больших данных		
1	Тема 1. Знакомство с Linux и его история. Различные дистрибутивы Linux и выбор подходящего для конкретной задачи. Установка Linux	Изучить материал: Совместное использование базы данных; Безопасность данных. Самостоятельно изучить и произвести базовую настройку Linux в частности: изменить обои рабочего стола, размер значков в Dock-панели, тему оформления и создать дополнительного пользователя с паролем. Тест: Знакомство с Linux
2	Тема 2. Основа работы с командной строкой Linux. Управление пользователями, группами и правами доступа	Изучить дополнительный материал по базовым командам Linux, приемам работы в командной строке Linux, терминалу Linux, а также управлению пользователями и группами и правам доступа к файлам и папкам Linux.

		Тест: Командная строка и ролевая модель Linux
3	Тема 3. Введение в пакетные менеджеры и установка пакетов для обработки больших данных. Работа со сторонними репозиториями	Изучить дополнительный материал: Популярные пакетные менеджеры Linux; Персональные архивы пакетов (PPA). Тест: Введение в пакетные менеджеры
4	Тема 4. Управление процессами и службами обработки больших данных. Мониторинг системы и журналы	Изучить дополнительный материал: Контроль нагрузки и процессов; 30 инструментов мониторинга системы Linux; Основы мониторинга системы; Log-файлы Linux. Тест: Управление и мониторинг процессов и служб
5	Тема 5. Основы безопасности больших данных. Настройка сети и брандмауэры	Изучить дополнительный материал: Базовая настройка безопасности Linux; Настройка брандмауэра в Ubuntu; Основы работы с SELinux. Тест: Основы безопасности больших данных
6	Тема 6. Создание и редактирование скриптов на Bash для обработки больших данных. Использование утилиты Cron для планирования задач	Изучить дополнительный материал: Основы Bash; Использование утилиты Cron. Тест: Bash и Cron

Рекомендуемый перечень вопросов для отработки в часы самостоятельной работы, подготовки к промежуточной аттестации:

1. Функции Архитектора данных.
2. Основы синтаксиса и структур в Python.
3. Использование библиотек Python.
4. Основные инструменты для работы с данными.
5. Математическая статистика.
6. Использование графиков при построении, отличие/название.
7. Знание анализа данных.
8. Технологии анализа данных.
9. Контейнеризация и управление контейнерами.
10. Работа с Linux.
11. Пакетные менеджеры Linux.
12. Безопасность больших данных.

Учебно-методическое обеспечение.

Обучающие материалы представлены в виде видеолекций, текстовых и графических материалов, размещенных на образовательной платформе <https://data.1t.ru/>.

Перечень основной и дополнительной учебной литературы.

Нормативно-правовые акты:

1. Национальная программа «Цифровая экономика Российской Федерации», утв. распоряжением Правительства Российской Федерации от 28 июля 2017 г. No 1632-р.
2. Федеральная проект «Искусственный интеллект» (паспорт) (утв. Президиумом Правительственной комиссии по цифровому развитию, использованию информационных технологий для улучшения качества жизни и условий ведения предпринимательской деятельности от 27.08.2020 № 17).

Основная литература:

1. Алекс Петров. Распределенные данные. Алгоритмы работы современных систем хранения информации / А. Петров – СПб: Питер, 2021. – 336 с.
2. Бураков, М.В. Системы искусственного интеллекта: Учебное пособие / М.В. Бураков. – М.: Проспект, 2019. – 440 с.
3. Карпова, И.П. Базы данных: Учебное пособие / И.П. Карпова. – Санкт-Петербург: Питер, 2021. – 240 с.
4. Кузнецов, В.А. Системный анализ, оптимизация и принятие решений: Учебник / В. А. Кузнецов, А. А. Черепяхин – Москва: КУРС, 2018. – 256 с.
5. Мартин Клеппман. Высоконагруженные приложения. Программирование, масштабирование, поддержка / М. Клеппман – СПб: Питер, 2018. – 740 с.
6. Мышев, А.В. Архитектура для интеллектуальных вычислительных и информационно-измерительных систем с нечеткой средой вычислений // Федеральный институт промышленной собственности, отделение ВПТБ – 2019. – 29 с.
7. Негус К. Библия Linux. 10-е издание / К. Негус. – Санкт-Петербург: Питер, 2022. – 928 с.
8. Немет, Эви, Снайдер, Гарт, Хейн, Трент, Уэйли, Бэн. Unix и Linux: руководство системного администратора. Том 1. 5-е изд. : Пер.с англ. — М.: ООО “Диалектика”, 2020. — 736 с.: ил.
9. Силен Д., Мейсман А., Али М. Основы Data Science и Big Data. Python и наука о данных / Силен Д., Мейсман А., Али М. ; пер. с англ. Матвеев Е. - СПб. : Питер, 2020. - 334 с. : ил. – (Библиотека программиста). - ISBN 978-5-4461-0944-9.

Дополнительная литература:

1. Брокман Д. Что мы думаем о машинах, которые думают: Ведущие мировые учёные об искусственном интеллекте / Пер. с англ. М. Исаков – М.: Альпина Паблишер, 2017. – 550 с.: ил.
 2. Бруссард М. Искусственный интеллект: Пределы возможного / Пер. с англ. Е. Арье – М.: Альпина Паблишер, 2020. – 362 с.
 3. Бычков А. Г. Сборник задач по теории вероятностей, математической статистике и методам оптимизации: учебное пособие / А.Г. Бычков. — Москва: Форум: ИНФРА-М, 2022. — 192 с. — (Среднее профессиональное образование).
 4. Душкин Р.В. Искусственный интеллект / Р.В. Душкин – М.: ДМК Пресс, 2022. – 380 с.: ил. – ISBN:978-5-97060-787-9.
 5. Мартин Ф. Архитекторы интеллекта: вся правда об искусственном интеллекте от его создателей / Пер. с англ. И. Ружмайкина. – СПб: Питер, 2019. – 400 с.: ил.
 6. О'Коннелл М. Искусственный интеллект и будущее человечества / Пер. с англ М. Кудряшова – М.: Бомбора, 2019. – 280 с.: ил.
 7. Поляков В. М. Методы оптимизации: учебное пособие / Поляков В. М., Агаларов З. С. — Москва: Дашков и К, 2022. — 86 с. — ISBN 978-5-394-05003-9.
 8. Рязанов С. И. Искусственный интеллект как множество - классификация искусственных интеллектов // Вузовская наука в современных условиях: сборник материалов 54-й Научно-технической конференции: в 3-х ч. Ульяновск: Ульяновский государственный технический университет, 2020. Ч. 1. С. 72-75.
 9. Фальк Ким. Рекомендательные системы на практике Практическое пособие / Пер. с англ. Д. М. Павлов. – Москва: ДМК Пресс, 2020. – 448 с.
- Учебно-методические и информационные материалы:
1. Асанов, В.Л. Стратегическое управление территориальным развитием — архитектурный менеджмент, администрирование: монография / В. Л. Асанов. — Москва: Издательство «Юрайт», 2023. — 275 с.
 2. Березкин Д. В., Рожнев А. Ю. Разработка модели оценки платежеспособности клиентов банка с применением алгоритмов машинного обучения / Березкин Д. В., Рожнев А. Ю. // Динамика сложных систем. – 2018. – Т. 12, № 4. - С. 59-66.
 3. Бизнес переходит на искусственный интеллект. / РБК+. Решения. #1 Искусственный интеллект, 5 декабря 2022. – [Электронный ресурс] – Режим доступа: <https://plus.rbc.ru/news/638ce98f7a8aa9f3126daaa2> (Дата обращения: 27.03.2024).
 4. Большие данные в социальных и гуманитарных науках: Сб. обзоров и рефератов / РАН. ИНИОН. Центр науч.-информ. исслед. по науке, образованию и

технологиям; отв. ред. – Гребенщикова Е.Г. – М., 2019. – 193 с. – (Сер.: Наука, образование и технологии). ISBN 978-5-248-00912-1.

5. Загорулько, Ю. А. Искусственный интеллект. Инженерия знаний: учебное пособие для вузов / Ю. А. Загорулько, Г. Б. Загорулько. — Москва: Издательство Юрайт, 2022. — 93 с. — (Высшее образование). — ISBN 978-5-534-07198-6.

6. Нагородская, В.Б. Новые технологии (блокчейн / искусственный интеллект) на службе права: научнометодическое пособие / под ред. Л. А. Новоселовой. — Москва: Проспект, 2019. — 128 с.

7. Расставить нейросети. / РБК+. Инновации. #1 Искусственный интеллект, 5 декабря 2022. — [Электронный ресурс] — Режим доступа: <https://plus.rbc.ru/news/638ce67f7a8aa9e27b22f26e> (Дата обращения: 27.03.2024).

8. Технологии позволяют учитывать специфику каждой отрасли / РБК+. От первого лица. #1 Искусственный интеллект, 5 декабря 2022. — [Электронный ресурс] — Режим доступа: <https://plus.rbc.ru/news/638ced6c7a8aa9e28f7bf148> (Дата обращения: 27.03.2024).

9. Хливненко Л. В., Пятакович Ф. А. Практика нейросетевого моделирования: учеб. пособие / Хливненко Л. В., Пятакович Ф. А. — СПб. Лань, 2019. - 196 с. - (Учебники для вузов. Специальная литература) (Бакалавриат и магистратура). — Библиогр.: с. 182-193. - ISBN 978-5-8114-3639-2.

10. Цифровая экономика от теории к практике: как российский бизнес использует искусственный интеллект [Электронный ресурс] / Исслед. РАЭК / НИУ ВШЭ при поддержке Microsoft. — 2019. — 66 с. — Режим доступа: <http://raec.ru/upload/files/190715-ii.pdf> (Дата обращения: 27.03.2024).

Информационное сопровождение.

Электронные образовательные ресурсы:

1. Бесплатные материалы по Data Engineering от преподавателей МФТИ // МФТИ. — [Электронный ресурс] — Режим доступа: <https://fpmi-edu.ru/free-de> (Дата обращения: 27.03.2024).

2. Бесплатный курс по Deep Learning от МФТИ // ФПМИ МФТИ. — [Электронный ресурс] — Режим доступа: <https://dls.samcs.ru> (Дата обращения: 27.03.2024).

3. Воронцов, К. В. Машинное обучение: курс лекций // MachineLearning.ru. — [Электронный ресурс] — Режим доступа: [http://www.recognition.su/wiki/index.php?title=Машинное_обучение_\(курс_лекций%2С_К.В.В_оронцов\)](http://www.recognition.su/wiki/index.php?title=Машинное_обучение_(курс_лекций%2С_К.В.В_оронцов)) (Дата обращения: 27.03.2024).

4. Курс «Big Data и Data Science: начни погружение с нуля» // Stepik. – [Электронный ресурс] – Режим доступа: <https://stepik.org/course/101687/promo> (Дата обращения: 27.03.2024).

5. Обучение Python и его библиотекам // W3Schools Spaces – [Электронный ресурс] – Режим доступа: <https://www.w3schools.com/python/default.asp> (Дата обращения: 27.03.2024).

6. Проектирование баз данных: распределенные базы и хранилища данных. Агрегирование // Национальный открытый университет «ИНТУИТ». – [Электронный ресурс] – Режим доступа: http://www.intuit.ru/studies/professional_retraining/953/courses/214/lecture/5508/ (Дата обращения: 27.03.2024).

7. Open Machine Learning Course / mlcourse.ai. – [Электронный ресурс]. – Режим доступа: <https://mlcourse.ai/book/index.html> (Дата обращения: 27.03.2024)

8. NLP-курс от ШАД // Lena Voita. – [Электронный ресурс] – Режим доступа: https://lena-voita.github.io/nlp_course.html (Дата обращения: 27.03.2024).

Электронные информационные ресурсы.

1. Искусственный интеллект // РБК – [Электронный ресурс]. – Режим доступа: <https://trends.rbc.ru/trends/tag/ai> (Дата обращения: 27.03.2024).

2. Курсы по искусственному интеллекту // УНТИ 2035– [Электронный ресурс] – Режим доступа: <https://ai.2035.university/> – (Дата обращения: 27.03.2024).

3. Курсы от образовательной платформы ООО «1Т» – [Электронный ресурс] – Режим доступа: <https://data.1t.ru/> (Дата обращения: 27.03.2024).

4. Национальный проект «Цифровая экономика» // Национальные проекты – [Электронный ресурс]. – Режим доступа: <https://национальныепроекты.рф/projects/tsifrovaya-ekonomika> (Дата обращения: 27.03.2024)

5. Федеральный проект «Искусственный интеллект»// Национальный проект «Цифровая экономика» // Национальные проекты – [Электронный ресурс]. – Режим доступа: <https://национальныепроекты.рф/projects/tsifrovaya-ekonomika/p-iskusstvennyu-intellekt-p> (Дата обращения: 27.03.2024).

Описание системы оценки качества освоения модуля 1.

Контроль результатов освоения модуля осуществляется в ходе текущего контроля успеваемости и промежуточной аттестации.

Текущий контроль предусмотрен в ходе изучения каждой темы.

Формами текущего контроля являются тесты и выполнение практических работ.

Тесты содержат не менее 5 вопросов с одним или несколькими правильными ответами. За каждый правильный ответ ставится 1 балл. Критерием прохождения теста является получение не менее 75% правильных ответов.

Примеры тестов.

Модуль 1. Базовый.

Раздел 1.2 Введение в системное администрирование больших данных.

Тема: Создание и редактирование скриптов на Bash для обработки больших данных.

Использование утилиты Cron для планирования задач.

Вопросы:

1. Как создать новый исполняемый скрипт на Bash?

a) new_script.sh

b) touch script.sh && chmod +x script.sh

c) nano script.sh

d) create_script script.sh

2. Какая команда используется для добавления шебанга (shebang) в начало скрипта?

a) #!/bash

b) #bin/bash

c) #!/bin/bash

d) #bash

3. Какие символы используются для комментариев в Bash-скрипте?

a) //

b) --

c) #

d) /* */

4. Как передать аргументы скрипту при его выполнении?

a) script.sh arg1 arg2

b) script.sh -a arg1 -b arg2

c) script.sh --argument1 arg1 --argument2 arg2

d) script.sh=arg1,arg2

5. Какая команда используется для редактирования cron-задач?

a) cron edit

b) crontab -e

c) cronedit

d) cron -e

6. Как задать cron-задачу, которая будет выполняться каждый день в 3 часа утра?

a) 3 * * * *

b) 0 3 * * *

c) * 3 * * *

d) @daily

7. Как удалить все cron-задачи для текущего пользователя?

a) cron -r

b) crontab -d

c) crontab -r

d) cron delete

8. Как просмотреть текущие cron-задачи для текущего пользователя?

a) crontab -l

b) cron view

c) cron list

d) list-cron

9. Какой символ используется для перенаправления вывода (stdout) в файл в Bash?

a) >

b) &>

c) <

d) |

10. Какой оператор используется для выполнения команды только в случае

успешного

выполнения предыдущей команды?

a) &&

b) ||

c) |

d) &

Выполнение практических работ оценивается в бинарной системе: зачтено / не зачтено.

Зачтено: задача решена, могут быть недочеты и неточности в решении.

Не зачтено: задача не решена.

Примеры практических работ.

Модуль 1. Базовый.

Раздел 1.1 Введение в архитектуру больших данных.

Тема 4. Введение в контейнеризацию с Docker. Управление контейнерами и создание образов, Docker-compose.

Задача: Практика работы с контейнерами.

Набор заданий подобный примеру, рассмотренному на практике:

1. Установить Docker, запустить контейнер hello-world, сделать скриншот.
2. Скачать и запустить контейнер Ubuntu 20.04, вывести в консоль версию ОС, сделать скриншот, удалить контейнер и образ, сделать скриншот.
3. Скачать и запустить контейнер nginx, опубликовать его, проверить доступность извне, сделать скриншот браузера, проверить в консоли версию установленного nginx и прочитать его логи, после обращения снаружи, сделать скриншот, удалить контейнер и образ, сделать скриншот.
4. Скачать и запустить приложение Grafana через Docker Compose, сделать скриншот compose файла, опубликовать его, сделать скриншот браузера с Grafana, сделать скриншот консоли с запущенными контейнерами, сделать скриншот сетей.

Создайте архив с созданными скриншотами в порядке их создания с указанием номеров заданий. Загрузите созданный файл LMS.

Промежуточная аттестация проводится в формате решения практико-ориентированной задачи (кейса).

Примеры практико-ориентированных задач (кейсов).

Модуль 1.

Задание 1: Анализ логов веб-сервера с использованием ELK Stack.

Цель: Настроить стек ELK для анализа логов веб-сервера и извлечения ценной информации из них.

Шаги:

1. Развернуть виртуальную машину с Linux.
2. Установить Docker и Docker Compose на виртуальную машину.
3. Настроить ELK Stack (Elasticsearch, Logstash, Kibana) для обработки и визуализации логов.
4. Создать файл конфигурации Logstash для чтения логов веб-сервера.
5. Настроить Elasticsearch для хранения данных.
6. Настроить Kibana для визуализации и анализа данных.

7. Настроить локальный веб-сервер (например, Apache или Nginx) для записи логов.
 8. Создать пайплайн Logstash для обработки логов веб-сервера.
 9. Создать дашборды в Kibana для отображения основных метрик.
 10. Провести анализ логов на предмет аномалий и угроз безопасности.
 11. Оценить производительность и эффективность стека ELK.
 12. Составить документацию о процессе настройки и использования ELK Stack.
- Ссылку на составленную документацию прислать в чат с преподавателем.

Задание 2: Мониторинг системных ресурсов с использованием Prometheus и Grafana.

Цель: Настроить систему мониторинга с использованием Prometheus и визуализацию данных с помощью Grafana для отслеживания системных ресурсов в реальном времени.

Шаги:

1. Развернуть виртуальную машину с Linux.
2. Установить Docker и Docker Compose на виртуальную машину.
3. Настроить Prometheus и экспортер для сбора метрик системных ресурсов.
4. Создать файл конфигурации Prometheus для определения целевых систем и портов для сбора метрик.
5. Настроить Grafana для визуализации данных из Prometheus.
6. Подключить Grafana к Prometheus и настроить источники данных.
7. Создать дашборды в Grafana для отображения основных метрик системных ресурсов (например, загрузка CPU, использование памяти, сетевой трафик и др.).
8. Настроить оповещения в Grafana для отправки уведомлений в случае превышения установленных порогов.
9. Запустить контейнеры Prometheus и Grafana с помощью Docker Compose.
10. Отследить изменения метрик в реальном времени через интерфейс Grafana.
11. Провести нагрузочное тестирование системы и оценить работу мониторинга.
12. Составить документацию о процессе настройки и использования Prometheus и Grafana.

Ссылку на составленную документацию прислать в чат с преподавателем.

Критерии и шкала оценивания:

- 0-4 балла: имеются содержательные и логические ошибки, решение кейса не найдено.
- 5-6 баллов: решение кейса в целом найдено, но оно неоптимально и/или имеются логические ошибки.

7-8 баллов: решение кейса найдено, но имеются неточности в решении.

9-10 баллов: решение кейса найдено, ошибки отсутствуют.

Максимально возможное число баллов за работу – 10.

Не менее 9 баллов – «отлично».

7-8 баллов – «хорошо».

5-6 баллов – «удовлетворительно».

0-4 балла – «неудовлетворительно».

4.2. Рабочая программа модуля 2. Профильный.

Цель освоения модуля 2 – приобретение слушателями знаний по основам хранения и обработки больших данных, изучение программных продуктов для работы с базами данных на основе технологий больших данных, машинного обучения и искусственного интеллекта; приобретение практического опыта по созданию и поддержке систем искусственного интеллекта, по проектированию и построению архитектур данных на основе нейросетевых моделей и методов.

Профессиональные компетенции, совершенствуемые и приобретаемые слушателями в процессе освоения модуля 2:

ПК-1.и. Способен применять методы и программные средства автоматизированного логического вывода и автоматизированной проверки гипотез.

ПК-4.р. Способен разрабатывать и применять методы машинного обучения для решения задач.

ПК-5.п. Способен использовать системы искусственного интеллекта на основе нейросетевых моделей и методов.

ПК-6.р. Способен создавать и поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов.

ПК-7.р. Способен осуществлять сбор и подготовку данных для систем искусственного интеллекта.

ПК-9.р. Способен создавать и внедрять одну или несколько сквозных цифровых субтехнологий искусственного интеллекта.

Планируемые результаты обучения по модулю 2.

По итогам освоения модуля слушатели должны:

Знать:

- методы и программный инструментарий для тестирования программных компонентов;
- инструменты и технологии, применяемые для реализации автоматизированного логического вывода и проверки гипотез в сфере анализа данных
- основные типы, источники больших данных, проблемы безопасности;
- методы и программный инструментарий технологий больших данных;
- базовые понятия и возможности теории игр для их применения в области машинного обучения и искусственного интеллекта;

- системы обработки и анализа больших массивов данных (SQL, NoSQL, Hadoop, SPARK и др.): принципы работы, функционал, язык запросов, преимущества и недостатки;
- технологии, методы и инструменты машинного обучения для решения профессиональных задач;
- нейронные сети: полносвязные, свёрточные и рекуррентные нейронные сети;
- методы обучения нейронных сетей;
- нейросетевые методы понижения размерности;
- архитектуры современных нейронных сетей и их использование для решения профессиональных задач
- методы, инструментальные средства и стандарты статистического и описательного анализа данных;
- основные платформы данных (облачные и внутрикорпоративные), применяемые при разработке решений на основе ИИ;
- основные виды представления данных: табличные, графовые, временные ряды;
- основные уровни представления данных;
- понятие сквозных цифровых субтехнологий искусственного интеллекта;
- методы и технологии внедрения сквозных цифровых субтехнологий искусственного интеллекта.

Уметь:

- применять методы машинного обучения для тестирования программных компонентов;
- проводить тестирование систем искусственного интеллекта;
- настраивать, конфигурировать и адаптировать программные средства автоматизированного логического вывода в сфере исследовательской деятельности;
- разрабатывать и настраивать алгоритмы автоматизированного логического вывода и проверки гипотез, учитывая специфику исследовательской деятельности;
- проектировать и настраивать конфигурации программных средств для эффективного применения методов автоматизированного логического вывода и проверки гипотез;
- проводить адаптацию и оптимизацию программных решений для учета требований и специфики конкретных задач;
- применять методы машинного обучения для работы с большими данными;
- применять методы машинного обучения и нейронных сетей;

– поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов

– внедрять технологии компьютерного зрения для обработки изображений и видео, задач биометрии;

– внедрять технологии обработки естественного языка для анализа и генерации текстов;

– внедрять технологии обучения с подкреплением для задач автоматизации производства и транспорта;

– использовать основные статистические методы анализа данных;

– моделировать данные в хранилищах (DWH, Data Lake, NoSQL хранилища данных);

Владеть:

– навыком построения полносвязной нейронной сети для задачи классификации;

– основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных;

– навыками использования различных инструментов и технологий в рамках процесса автоматизации логического вывода и проверки гипотез;

– способами улучшить качество модели с помощью методов Feature engineering;

– навыком использования библиотеки PyTorch, применения GPU в PyTorch;

– навыком построения полносвязной нейронной сети для задачи классификации;

– основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных;

– методами машинного обучения и нейронных сетей;

– навыком создания логической и физической базы данных;

– навыком BI-аналитики;

– языками запросов для обращения к СУБД;

– навыком работы с генеративными моделями GAN для задач генерации;

– навыком работы с автокодировщиками для задач уменьшения размерности и поиска аномалий;

– навыком работы с технологией RL – обучения с подкреплением.

Учебно-тематический план модуля 2 Профильный.

№ п/п	Наименование дисциплины, модуля, темы	Трудо-емкость		В том числе				Самостоятельная работа	Форма контроля
		В зачетных единицах	В часах	Контактная работа ¹					
				Всего	из них	Лекции	Практические занятия		
1	2	3	4	5	6	7	8	9	
Раздел 2.1 Проектирование архитектур: большие данные, машинное обучение, искусственный интеллект									
1	Тема 1. Введение в базы данных и теорию хранения данных. Реляционные базы данных, использование SQL		8	6	2	4	2	тест, практическая работа	
2	Тема 2. Введение в Hadoop. DWH, Data Lake, NoSQL хранилища данных		8	6	2	4	2	тест, практическая работа	
3	Тема 3. Массово параллельная обработка и анализ данных Работа с PySpark. Поточковая обработка данных (data streaming, event processing). Шины данных (kafka)		8	6	2	4	2	тест, практическая работа	
4	Тема 4. Введение в ML. Основные задачи, виды классических моделей, метрики качества моделей. VI-системы и визуализация данных		8	6	2	4	2	тест, практическая работа	
5	Тема 5. Погружение в ML. Обучение с учителем		10	8	2	6	2	тест, практическая работа	
6	Тема 6. Погружение в ML. Обучение без учителя		10	8	2	6	2	тест, практическая работа	
7	Тема 7. Введение в нейронные сети. Процесс, стадии и методологии разработки решений на основе искусственного интеллекта		8	6	2	4	2	тест, практическая работа	
8	Тема 8. CV - компьютерное зрение		10	6	2	4	4		
9	Тема 9. NLP – обработка естественного языка		10	6	2	4	4	тест, практическая работа	
Раздел 2.2. Построение архитектур больших данных									

¹ С применением дистанционных образовательных технологий и электронного обучения

10	Тема 1. Введение в Kubernetes и установка Minikube. Управление ресурсами архитектур больших данных в Kubernetes		12	8	2	6	4	тест, практическая работа
11	Тема 2. Масштабирование и автомасштабирование архитектур больших данных. Мониторинг в Kubernetes		12	8	2	6	4	тест, практическая работа
12	Тема 3. Управление пакетами обработки больших данных с Helm		6	4	1	3	2	тест, практическая работа
13	Тема 4. CI/CD пакетов обработки больших данных в Kubernetes. Основы CI/CD на примере Gitlab CI		16	8	2	6	8	тест, практическая работа
14	Тема 5. DataOps в Kubernetes с Apache Airflow		6	4	1	3	2	тест, практическая работа
15	Тема 6. MLOps в Kubernetes с Kubeflow		6	4	1	3	2	тест, практическая работа
16	Промежуточная аттестация		4	4		4		Решение кейса
17	Итого по модулю 2		142	98	27	71	44	Решение кейсов

Раздел 2.1. Проектирование архитектур: большие данные, машинное обучение, искусственный интеллект.

Тема 1. Введение в базы данных и теорию хранения данных. Реляционные базы данных, использование SQL.

Типы больших данных: метаданные, полуструктурированные, структурированные, неструктурированные. Виды источников данных: созданные человеком, созданные машинами. Источники информации, в том числе информации, необходимой для обеспечения деятельности в предметной области заказчика исследования. Основные цели моделирования данных. Реляционные базы данных для хранения структурированных данных. Сравнение OLAP и OLTP систем. Создание таблиц нормализованных данных. Внедрение денормализованных схем (например, STAR, Snowflake). Использование реляционных баз данных SQL: GreenPlum, Postgres, Oracle. Использование SQL в Big Data.

Тема 2. Введение в Hadoop. DWH, Data Lake, NoSQL хранилища данных.

Основы Hadoop. Архитектура Apache Hadoop. Архитектура HDFS. Структура файлов. Алгоритмы чтения и записи в HDFS. Обзор API для работы с HDFS. Object Storage, сходство и отличия от HDFS. Введение в хранилища данных. Архитектура хранилища данных. Хранилища данных NoSQL, назначение и особенности. Отличия от реляционных баз данных. NoSQL в BigData. Рекомендации по использованию и опыт использования

разнородных источников данных и информации в задачах анализа больших данных. Принципы обеспечения безопасности данных в облачных хранилищах данных и нереляционных базах данных.

Тема 3. Массово-параллельная обработка и анализ данных Работа с PySpark. Поточковая обработка данных (data streaming, event processing). Шины данных (kafka).

Массово параллельные базы данных в BigData. Современная технологическая инфраструктура высокопроизводительных и распределенных вычислений. Технология MapReduce. Работа с распределенной кластерной системой. Системы обмена сообщениями Storm и Kafka. Создание конвейеры данных с помощью Storm и Kafka. Шины данных Kafka.

Тема 4. Введение в ML. Основные задачи, виды классических моделей, метрики качества моделей. BI-системы и визуализация данных.

Машинное обучение на больших данных. Машинное обучение: классификация, кластеризация, обнаружение выбросов, фильтрация. Методы и модели классификации: логистическая регрессия, деревья решений. Методы и модели регрессии: линейная регрессия, деревья решений. Метрики качества моделей. Что такое BI-системы и зачем нужны. Обзор популярных решений. Введение в Superset. Разработка простого дашборда данных.

Тема 5. Погружение в ML. Обучение с учителем.

Обучение с учителем: задача классификации, метод опорных векторов, решающие деревья, случайный лес, K-ближайших соседей (KNN), бустинги. Принципы работы таких моделей. Кросс-валидация.

Тема 6. Погружение в ML. Обучение без учителя.

Обучение без учителя: метод наименьших квадратов, задача кластеризации данных, поиск аномалий.

Тема 7. Введение в нейронные сети. Процесс, стадии и методологии разработки решений на основе искусственного интеллекта.

Введение в нейронные сети. Алгоритм обратного распространения ошибки для обучения нейронных сетей. Основные архитектуры и задачи нейронных сетей. Производная, оптимизация. Разработка и согласование технического задания на создание методической и технологической инфраструктуры больших данных. Управление архитектурой организации. Методы управления жизненным циклом информационно-технологической инфраструктуры организации. Методы управления проектами создания информационно-технологической инфраструктуры организации. Инструменты и методы согласования с заказчиками требований к результатам аналитических исследований с использованием технологий

больших данных. Архитектура и принципы работы промышленных решений, созданных на основе ИИ. Области применения искусственного интеллекта.

Тема 8. CV - компьютерное зрение.

Классификация, сегментация изображений. Распознавание объектов на изображении. Распознавание текста на изображении. Способы оптимизации. Анализ изображений и видео с помощью методов искусственного интеллекта.

Тема 9. NLP – обработка естественного языка.

Основные задачи и для чего используется. Что такое векторное представление слов, мешок слов и N-граммы, TF-IDF, эмбендинги. Виды нейронных сетей для задач NLP. Использование BERT для построения эмбендинга слов. Анализ естественного языка с помощью методов искусственного интеллекта.

Раздел 2.2. Построение архитектур больших данных.

Тема 1. Введение в Kubernetes и установка Minikube. Управление ресурсами архитектур больших данных в Kubernetes.

Обзор Kubernetes и его роли в оркестрации контейнеров. Установка и конфигурация Minikube для локального развертывания кластера Kubernetes для обработки больших данных. Основы управления ресурсами (CPU, память) в Kubernetes. Запуск приложений больших данных с определением требуемых ресурсов (например, использование Deployment и указание ресурсов в манифестах).

Тема 2. Масштабирование и автомасштабирование архитектур больших данных. Мониторинг в Kubernetes.

Понятие масштабирования и автомасштабирования в Kubernetes при обработке больших данных. Использование Horizontal Pod Autoscaler (HPA) для автоматического масштабирования приложений на основе нагрузки. Основы мониторинга состояния кластера и приложений больших данных в Kubernetes. Использование инструментов мониторинга, таких как Prometheus и Grafana, для сбора и визуализации метрик.

Тема 3. Управление пакетами обработки больших данных с Helm.

Введение в Helm и его роль в управлении пакетами обработки больших данных для Kubernetes. Установка и настройка Helm в кластере Kubernetes. Создание и управление пакетами Helm для развертывания приложений и сервисов больших данных в Kubernetes.

Тема 4. CI/CD пакетов обработки больших данных в Kubernetes. CI/CD на примере Gitlab CI.

Определение основных понятий CI/CD. Знакомство с GitLab CI. Настройка проекта в GitLab CI. Интеграция тестирования и сборки. Роль CI/CD в разработке и развертывании приложений больших данных в Kubernetes. Использование GitLab CI для автоматизации процессов сборки, тестирования и развертывания приложений в Kubernetes.

Тема 5. DataOps в Kubernetes с Apache Airflow.

Введение в Apache Airflow и его роль в автоматизации процессов DataOps в Kubernetes. Установка Apache Airflow в Kubernetes и его настройка. Создание и запуск DAG (Directed Acyclic Graph) для обработки данных в Kubernetes с использованием Apache Airflow.

Тема 6. MLOps в Kubernetes с Kubeflow.

Введение в Kubeflow и его роль в автоматизации процессов MLOps в Kubernetes. Установка Kubeflow в Kubernetes и его настройка. Создание и управление Kubeflow Pipelines для развертывания и обучения моделей машинного обучения в Kubernetes.

Содержание практических занятий по модулю 2.

№ темы	Наименование (содержание) темы, по которой предусмотрено занятие семинарского типа	Формы и методы проведения
Раздел 2.1. Проектирование архитектур: большие данные, машинное обучение, искусственный интеллект		
1	Создать Логическую и физическую модель базы данных продуктового магазина. Установить базу данных PostgreSQL и PGAdmin. Создать базу данных продуктового магазина. Провести ее к 3NF, при необходимости провести ее денормализацию	Решение практических заданий
2	Произвести установку экосистемы Hadoop на виртуальную машину или использовать Docker. Работа с HDFS (интерфейс командной строки). Произвести развертывание окружения: СУБД + dbt. Конфигурация проекта dbt, установка модуля dbtVault. Выполнить проектирование детального слоя на базе подхода Data Vault, подготовка метаданных для кодогенерации. Автоматизировать наполнения детального слоя данных с помощью dbt + dbtVault. Сформировать витрины данных на основе детального слоя	Решение практических заданий
3	Массово-параллельные базы данных в BigData. Современная технологическая инфраструктура высокопроизводительных и распределенных вычислений. Технология MapReduce. Работа с распределенной кластерной системой.	Решение практических заданий

	Системы обмена сообщениями Storm и Kafka. Создание конвейеры данных с помощью Storm и Kafka. Шины данных Kafka	
4	Создание собственной модели линейной регрессии, сравнение работы с готовым решением в scikit-learn. Добавление Superset в DWH. Создать полноценный DWH с Postgres и Superset'ом	Решение практических заданий
5	Предобработка данных, построение нескольких классических моделей и выбор лучшей модели на данных	Решение практических заданий
6	Поиск аномалий в данных, сегментация PCA. Уменьшение размерности данных	Решение практических заданий
7	Построить собственную модель линейной регрессии на основе библиотеки PyTorch. Провести классификацию знаний. Изучить заданную предметную область и построить модель знаний в виде графа	Решение практических заданий
8	Построение сверточной нейронной сети для задачи многоклассовой классификации	Решение практических заданий
9	Анализ естественного языка с помощью методов искусственного интеллекта. Воспользоваться TF-IDF или эмбендингами для анализа токсичности текстов	Решение практических заданий
Раздел 2.2. Построение архитектур больших данных		
1	Установить и настроить Minikube. Запустить приложения для обработки больших данных в Kubernetes с ограничением ресурсов	Решение практических заданий
2	Настройка НРА на основе используемых ресурсов и метрик, таких как нагрузка CPU. Установка и настройка Prometheus и Grafana для мониторинга кластера Kubernetes и его приложений обработки больших данных	Решение практических заданий
3	Установка Helm, создание и управление пакетами для приложений больших данных в Kubernetes	Решение практических заданий
4	Настройка и автоматизация процесса CI/CD для приложений больших данных в Kubernetes с использованием GitLab CI	Решение практических заданий
5	Создание DAG для автоматизации процесса обработки данных в Kubernetes с использованием Apache Airflow	Решение практических заданий
6	Создание Kubeflow Pipeline для развертывания и обучения модели машинного обучения в Kubernetes	Решение практических заданий

Содержание самостоятельной работы слушателей по модулю 2.

Основная цель самостоятельной работы слушателей – закрепление знаний, полученных в ходе лекционных и практических занятий.

Индивидуальная консультационная работа преподавателей со слушателями осуществляется весь период обучения.

№ темы	Наименование (содержание) темы, по которой предусмотрена самостоятельная работа	Формы и методы проведения
Раздел 2.1. Проектирование архитектур: большие данные, машинное обучение, искусственный интеллект		
1	Тема 1. Введение в базы данных и теорию хранения данных. Реляционные базы данных, использование SQL	Изучить материал: Совместное использование базы данных. Безопасность данных. Функционал PGAdmin для PostgreSQL. Тест: Введение в базы данных и теорию хранения данных. Реляционные базы данных
2	Тема 2. Введение в Hadoop. DWH, Data Lake, NoSQL хранилища данных	Изучить материал: Распределенные файловые системы. Их составляющие. Их достоинства, недостатки и сфера применения. Документно-ориентированная модель данных MongoDB. Изучить принципы моделирования данных в Data Warehouse. Тест: Hadoop. Хранилища данных
3	Тема 3. Массово параллельная обработка и анализ данных Работа с PySpark. Поточковая обработка данных (data streaming, event processing). Шины данных (kafka)	Изучить принципы обеспечения безопасных вычислений в распределенных вычислительных средах, а также архитектуру обработки данных в реальном времени – Apache Kafka. Тест: Массово параллельная обработка и анализ данных. Конвейер данных
4	Тема 4. Введение в ML. Основные задачи, виды классических моделей, метрики качества моделей. VI-системы и визуализация данных	Изучить материал: Первые шаги в VI-аналитике. Роль Архитектора данных. Документация scikit-learn. Тест: Введение в ML. VI-системы и визуализация данных
5	Тема 5. Погружение в ML. Обучение с учителем	Изучить материал: Принцип работы Наивного Байесовского классификатора. Тест: Обучение с учителем
6	Тема 6. Погружение в ML. Обучение без учителя	Изучить материал: • Восстановление регрессии данных • Иерархическая кластеризация • Модельная кластеризация • Сеточная кластеризация Тест: Обучение без учителя
7	Тема 7. Введение в нейронные сети. Процесс, стадии и методологии разработки решений на основе искусственного интеллекта	Изучить материал: Использование GPU в PyTorch. Библиотека TensorFlow и основные различия. Технологии межличностной и групповой коммуникации в деловом взаимодействии, основы

		конфликтологии. Современный опыт использования анализа больших данных. Тест: Нейронные сети. Роль искусственного интеллекта в бизнесе
8	Тема 8. CV - компьютерное зрение	Изучить тему: Использование CV в видео. Тест: Computer vision - основы
9	Тема 9. NLP – обработка естественного языка	Изучить материал: Модели трансформеров и что такое self-attention. Тест: Основные понятия NLP
Раздел 2.2. Построение архитектур больших данных		
1	Тема 1. Введение в Kubernetes и установка Minikube. Управление ресурсами архитектур больших данных в Kubernetes	Изучить документации Kubernetes и Minikube для более глубокого понимания их функциональности и возможностей. Протестируйте с запуском различных приложений больших данных в кластере Kubernetes с разными настройками ресурсов и ограничений. Тест: Введение в Kubernetes и управление ресурсами
2	Тема 2. Масштабирование и автомасштабирование архитектур больших данных. Мониторинг в Kubernetes	Изучите официальную документацию Kubernetes о масштабировании и автомасштабировании приложений с основными концепциями и методами работы Horizontal Pod Autoscaler (HPA). Протестируйте с различными настройками мониторинга и масштабирования в вашем кластере Kubernetes. Изучите особенности работы и возможности каждого инструмента. Проанализируйте полученные данные мониторинга и их влияние на масштабирование приложений. Оптимизируйте процессы и настройки для повышения эффективности обработки больших данных в Kubernetes. Тест: Масштабирование и мониторинг в Kubernetes
3	Тема 3. Управление пакетами обработки больших данных с Helm	Изучите официальную документацию по Helm для полного понимания его роли в управлении пакетами в Kubernetes. Дополнительно попрактикуйтесь в создании собственных пакетов Helm и их настройке для развертывания приложений и сервисов обработки больших данных в Kubernetes. Тест: Пакеты обработки больших данных
4	Тема 4. CI/CD пакетов обработки больших данных в Kubernetes. Основы CI/CD на примере Gitlab CI	Проведите эксперименты с различными конфигурациями пайплайнов в GitLab CI, включая развертывание на различные окружения Kubernetes. Попробуйте

		интегрировать дополнительные инструменты и сервисы в свои CI/CD процессы, такие как статический анализ кода, тестирование безопасности и другие. Разработайте собственные скрипты и шаги для вашего CI/CD пайплайна, учитывая особенности и требования вашего проекта. Подключите мониторинг и логирование ваших CI/CD процессов для отслеживания и анализа их производительности и эффективности. Тест: Основы CI/CD
5	Тема 5. DataOps в Kubernetes с Apache Airflow	Попробуйте создать более сложные DAG (Directed Acyclic Graph) для обработки данных, включая различные типы операций, планирование и управление зависимостями. Проведите эксперименты с настройкой планировщика и выполнения задач в Apache Airflow. Изучите возможности мониторинга и управления выполнением задач в Apache Airflow, включая интеграцию с инструментами мониторинга и логирования. Попробуйте интегрировать Apache Airflow с другими инструментами и сервисами для создания полноценного DataOps процесса в Kubernetes. Тест: DataOps в Kubernetes с Apache Airflow
6	Тема 6. MLOps в Kubernetes с Kubeflow	Изучите примеры Kubeflow Pipelines для различных задач машинного обучения. Проведите собственные эксперименты с настройкой и оптимизацией Kubeflow в среде Kubernetes. Изучите интеграцию Kubeflow с другими инструментами и платформами для управления процессами машинного обучения. Тест: MLOps в Kubernetes с Kubeflow

Рекомендуемый перечень вопросов для отработки в часы самостоятельной работы, подготовки к промежуточной аттестации.

1. Организация реляционных баз данных.
2. Особенности хранения данных.
3. Принципы параллельной обработки данных.
4. Модели машинного обучения.
5. Обучение с учителем.

6. Обучение без учителя.
7. Нейронные сети.
8. Области применения искусственного интеллекта.
9. Роль искусственного интеллекта в бизнесе.
10. Принятие бизнес-решений на основе данных.
11. Computer vision – основы.
12. Основные понятия NLP.
13. Инструментальные средства автоматизации развёртывания, масштабирования и координации контейнеризованными приложениями.
14. Основы обработки больших данных на базе платформы Kubernetes.
15. Инструментами и платформами для управления процессами машинного обучения.

Учебно-методическое обеспечение.

Обучающие материалы представлены в виде видеолекций, текстовых и графических материалов, размещенных на образовательной платформе <https://data.1t.ru/>.

Перечень основной и дополнительной учебной литературы.

Нормативно-правовые акты:

1. Национальная программа «Цифровая экономика Российской Федерации», утв. распоряжением Правительства Российской Федерации от 28 июля 2017 г. № 1632-р.
2. Федеральный проект «Искусственный интеллект» (паспорт) (утв. президиумом Правительственной комиссии по цифровому развитию, использованию информационных технологий для улучшения качества жизни и условий ведения предпринимательской деятельности от 27.08.2020 № 17).

Основная литература:

1. Алекс Петров. Распределенные данные. Алгоритмы работы современных систем хранения информации – СПб: Питер, 2021. – 336 с.
2. Бураков, М.В. Системы искусственного интеллекта: Учебное пособие / М.В. Бураков. – М.: Проспект, 2019. – 440 с.
3. Маркин, А. В. Программирование на SQL: учебное пособие для среднего профессионального образования / А. В. Маркин. — Москва: Издательство Юрайт, 2020. — 435 с. — (Профессиональное образование). — ISBN 978-5-534-11093-7. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/456926>
4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2: учебник и практикум для вузов / А. В. Маркин. — 2-е изд., испр. и доп. — Москва: Издательство Юрайт, 2020. —

340 с. — (Высшее образование). — ISBN 978-5-534-12258-9. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/451185>.

5. Мартин Ф. Архитекторы интеллекта: вся правда об искусственном интеллекте от его создателей / Пер. с англ. И. Рузмайкина. — СПб: Питер, 2019. — 400 с.: ил.

6. Мышев, А.В. Архитектура для интеллектуальных вычислительных и информационно-измерительных систем с нечеткой средой вычислений — 2019.

7. Нархид Ния, Шапира Гвен, Палино Тодд. Apache Kafka. Поточковая обработка и анализ данных — СПб: Питер, 2021. — 320 с.: ил.

Дополнительная литература:

1. Болотова, Ю.А. Методы и алгоритмы интеллектуальной обработки цифровых изображений: учебное пособие / Ю. А. Болотова, А. А. Друки, В. Г. Спицын; Национальный исследовательский Томский политехнический университет (ТПУ). — Томск: Изд-во ТПУ, 2016. — 207 с.: ил. — Библиогр.: с. 206. — ISBN 978-5-4387-0710-3.

2. Брокман Д. Что мы думаем о машинах, которые думают: Ведущие мировые учёные об искусственном интеллекте / Пер. с англ. М. Исаков — М.: Альпина Паблишер, 2017. — 550 с.: ил.

3. Бруссард М. Искусственный интеллект: Пределы возможного / Пер. с англ. Е. Арье — М.: Альпина Паблишер, 2020. — 362 с.

4. Брэдшоу Шеннон, Брэзил Йон, Ходоров Кристина. MongoDB: полное руководство. Мощное и масштабируемая система управления базами данных. 3-я редакция / Пер. с англ. Д. А. Беликова. — М.: ДМК Пресс, 2020. - 540 с.: ил.

5. Душкин Р.В. Искусственный интеллект / Р.В. Душкин — М.: ДМК Пресс, 2022. — 380 с.: ил. — ISBN:978-5-97060-787-9.

6. Маркус Г. Искусственный интеллект: Перезагрузка. Как создать машинный разум, которому действительно можно доверять / Г. Маркус, Э. Дэвис. — М.: Альпина Паблишер, 2021. — 328 с.

7. Мартин Ф. Архитекторы интеллекта: вся правда об искусственном интеллекте от его создателей / Пер. с англ. И. Рузмайкина. — СПб: Питер, 2019. — 400 с.: ил.

8. О'Коннелл М. Искусственный интеллект и будущее человечества / Пер. с англ. М. Кудряшова — М.: Бомбора, 2019. — 280 с.: ил.

9. Пиковер К. Искусственный интеллект. Иллюстрированная история. От автоматов до нейросетей / Пер. с англ. А. Ефримова. — М.: Синдбад, 2022. — 250 с.

10. Поляков В. М. Методы оптимизации: учебное пособие / Поляков В. М., Агаларов З. С. — Москва: Дашков и К, 2022. — 86 с. — ISBN 978-5-394-05003-9.

11. Рязанов С. И. Искусственный интеллект как множество - классификация искусственных интеллектов // Вузовская наука в современных условиях: сборник материалов 54-й Научно-технической конференции: в 3-х ч. – Ульяновск: Ульяновский государственный технический университет, 2020. Ч. 1. – 72-75 с.

12. Сергеев, Л. И. Цифровая экономика: учебник для вузов / Л. И. Сергеев, Д. Л. Сергеев, А. Л. Юданова ; под редакцией Л. И. Сергеева. — 2-е изд., перераб. и доп. — М.: Издательство Юрайт, 2023. — 437 с. — (Высшее образование). — ISBN 978-5-534-15797-0.

13. Томас Дэвенпорт. Внедрение искусственного интеллекта в бизнес-практику. Преимущества и сложности / Пер. с англ. З. Мамедьяров. – М.: Альпина Паблишер, 2021. – 316 с.

14. Фальк Ким. Рекомендательные системы на практике Практическое пособие / Пер. с англ. Д. М. Павлов. – Москва: ДМК Пресс, 2020. – 448 с.

15. Цзэн Мин. Как Alibaba использует искусственный интеллект в бизнесе: Сетевое взаимодействие и анализ данных / Пер. с кит. В. Ионов. – М: Альпина Паблишер, 2022. – 360 с.: ил.

Учебно-методические и информационные материалы.

1. Асанов, В. Л. Стратегическое управление территориальным развитием — архитектурный менеджмент, администрирование: монография / В. Л. Асанов. — Москва: Издательство Юрайт, 2023. — 275 с. — (Актуальные монографии). — ISBN 978-5-534-12772-0.

2. Большие данные в социальных и гуманитарных науках: Сб. обзоров и рефератов / РАН. ИНИОН. Центр науч.-информ. исслед. по науке, образованию и технологиям; отв. ред. – Гребенщикова Е.Г. – М., 2019. – 193 с. – (Сер.: Наука, образование и технологии). ISBN 978-5-248-00912-1.

3. Бизнес переходит на искусственный интеллект. / РБК+. Решения. #1 Искусственный интеллект, 5 декабря 2022. – [Электронный ресурс] – Режим доступа: <https://plus.rbc.ru/news/638ce98f7a8aa9f3126daaa2> (Дата обращения: 27.03.2024).

4. Загорулько, Ю. А. Искусственный интеллект. Инженерия знаний: учебное пособие для вузов / Ю. А. Загорулько, Г. Б. Загорулько. — Москва: Издательство Юрайт, 2022. — 93 с. — (Высшее образование). — ISBN 978-5-534-07198-6.

5. Нагородская, В.Б. Новые технологии (блокчейн / искусственный интеллект) на службе права: научнометодическое пособие / под ред. Л. А. Новоселовой. — Москва: Проспект, 2019. — 128 с.

6. Расставить нейросети. / РБК+. Инновации. #1 Искусственный интеллект, 5 декабря 2022. – [Электронный ресурс] – Режим доступа: <https://plus.rbc.ru/news/638ce67f7a8aa9e27b22f26e> (Дата обращения: 27.03.2024).

7. Технологии позволяют учитывать специфику каждой отрасли / РБК+. От первого лица. #1 Искусственный интеллект, 5 декабря 2022. – [Электронный ресурс] – Режим доступа: <https://plus.rbc.ru/news/638ced6c7a8aa9e28f7bf148> (Дата обращения: 27.03.2024).

8. Цифровая экономика от теории к практике: как российский бизнес использует искусственный интеллект [Электронный ресурс] / Исслед. РАЭК / НИУ ВШЭ при поддержке Microsoft. – 2019. – 66 с. – Режим доступа: <http://raec.ru/upload/files/190715-ii.pdf> (Дата обращения: 27.03.2024).

Статьи:

1. Алейникова, Ю.В.; Матвеев, В.В. Цифровая экосистема. Анализ применения искусственного интеллекта // Здоровье – основа человеческого потенциала: проблемы и пути их решения. – 2020. – Т. 15. – № 3. – С. 1480-1487.

2. Безгачев, Ф.В. Особенности обеспечения кибербезопасности посредством технологий искусственного интеллекта / В сборнике: Актуальные проблемы борьбы с преступностью: вопросы теории и практики. Материалы XXV международной научно-практической конференции, в 2-х частях. – Красноярск, 2022. – С. 43-44.

3. Городнова, Н.В. Применение искусственного интеллекта в бизнес-сфере: современное состояние и перспективы // Вопросы инновационной экономики. – 2021. – Том 11. – № 4. – С. 1473-1492. – doi: 10.18334/vines.11.4.112249.

4. Дин Но., Афанасьев Г.И. Состояние и перспективы применения искусственного интеллекта в визуализирующей диагностике заболеваний легких // E-Scio. – 2022. – № 4 (67). – С. 653-664.

5. Доржиева, В.В. Цифровизация промышленности: роль искусственного интеллекта и возможности для России // Вопросы инновационной экономики. – 2022. – Т. 12. № 4. – С. 2383-2394.

6. Коженков, А.О.; Середа, А.М.; Поливаев, О.И. Искусственный интеллект в современном автомобилестроении / В сборнике: Теория и практика инновационных технологий в АПК. Материалы национальной научно-практической конференции. – Воронеж, 2022. – С. 210-216.

7. Красов, А.В.; Штеренберг, С.И.; Фахрутдинов, Р.М.; Рыжаков, Д.В.; Пестов, И.Е. Анализ информационной безопасности предприятия на основе сбора данных пользователей с открытых ресурсов и мониторинга информационных ресурсов с

использованием машинного обучения [Электронный ресурс] // T-Comm. 2018. №10. – Режим доступа: <https://cyberleninka.ru/article/n/analiz-informatsionnoy-bezopasnosti-predpriyatiya-na-osnove-sbora-dannyh-polzovateley-s-otkrytyh-resursov-i-monitoringa> (Дата обращения: 27.03.2024).

8. Куницкая, О.М. Правовое регулирование искусственного интеллекта применительно к задачам развития экономики / В сборнике: Право и инновации: новые вызовы технологической революции. Материалы II Приволжского юридического конгресса. – Уфа, 2022. – С. 77-91.

9. Лазарева, М.М.; Калюкарин, А.В. Преимущества искусственного интеллекта. Применение в банковской сфере // Вопросы устойчивого развития общества. – 2022. – № 9. – С. 312-316.

10. Лазько, Н.В. Применение искусственного интеллекта в нефтегазовой отрасли // Мехатроника, автоматика и робототехника. – 2023. – № 11. – С. 155-158.

11. Львович, И.Я. Проблемы методологии проектирования интеллектуальных информационных систем // Информационные технологии в управлении, автоматизации и мехатронике. – 2020. – С. 120-123.

12. Машошин, А.И. Применение искусственного интеллекта при создании систем управления силами ВМФ // Морская радиоэлектроника – 2022. – № 2 (80). – С. 22-25.

13. Нарушения при работе с информационными системами россельхознадзора будет выявлять искусственный интеллект // Пищевая промышленность. – 2022. – № 3. – С. 76.

14. Пиляй, И.В. Применение систем искусственного интеллекта для оценки времени и стоимости строительного проекта // Строительство и архитектура. – 2023. – Т. 11. – № 1. – С. 19.

15. Репин, И. С. Перспективные направления использования искусственного интеллекта в оперативно-розыскной деятельности // Журнал правовых и экономических исследований. – 2022. – № 2. – С. 125-130.

16. Субботин, А.В.; Тагирова, Л.Ф. Математическое моделирование информационных процессов проектирования интеллектуальных систем на основе использования метода Мамдани // Информационные технологии как основа прогрессивных научных исследований. – 2020. – С. 95-99.

17. Шиллер, А.В. Место этической системы в архитектуре искусственного интеллекта // Вестник Томского государственного университета. – 2020. – №. 456. – С. 99-103.

Информационное сопровождение.

Электронные образовательные ресурсы:

1. Бесплатные материалы по Data Engineering от преподавателей МФТИ // МФТИ. – [Электронный ресурс] – Режим доступа: <https://fpmi-edu.ru/free-de> (Дата обращения: 27.03.2024).
2. Бесплатный курс по Deep Learning от МФТИ // ФПМИ МФТИ. – [Электронный ресурс] – Режим доступа: <https://dls.samcs.ru> (Дата обращения: 27.03.2024).
3. Воронцов, К. В. Машинное обучение: курс лекций // MachineLearning.ru. – [Электронный ресурс] – Режим доступа: [http://www.recognition.su/wiki/index.php?title=Машинное_обучение_\(курс_лекций%2C_К.В.Воронцов\)](http://www.recognition.su/wiki/index.php?title=Машинное_обучение_(курс_лекций%2C_К.В.Воронцов)) (Дата обращения: 27.03.2024).
4. Курс «Big Data и Data Science: начни погружение с нуля» // Stepik. – [Электронный ресурс] – Режим доступа: <https://stepik.org/course/101687/promo> (Дата обращения: 27.03.2024).
5. Обучение Python и его библиотекам // W3Schools Spaces – [Электронный ресурс] – Режим доступа: <https://www.w3schools.com/python/default.asp> (Дата обращения: 27.03.2024).
6. Проектирование баз данных: распределенные базы и хранилища данных. Агрегирование // Национальный открытый университет «ИНТУИТ». – [Электронный ресурс] – Режим доступа: http://www.intuit.ru/studies/professional_retraining/953/courses/214/lecture/5508/ (Дата обращения: 27.03.2024).
7. Open Machine Learning Course / mlcourse.ai. – [Электронный ресурс]. – Режим доступа: <https://mlcourse.ai/book/index.html> (Дата обращения: 27.03.2024)
8. NLP-курс от ШАД // Lena Voita. – [Электронный ресурс] – Режим доступа: https://lena-voita.github.io/nlp_course.html (Дата обращения: 27.03.2024).

Электронные информационные ресурсы:

1. Искусственный интеллект // РБК – [Электронный ресурс]. – Режим доступа: <https://trends.rbc.ru/trends/tag/ai> (Дата обращения: 27.03.2024).
2. Курсы по искусственному интеллекту // УНТИ 2035– [Электронный ресурс]. – Режим доступа: <https://ai.2035.university/> – (Дата обращения: 27.03.2024).
3. Курсы от образовательной платформы ООО «1Т» – [Электронный ресурс]. – Режим доступа: <https://data.1t.ru/> (Дата обращения: 27.03.2024).

4. Национальный проект «Цифровая экономика» // Национальные проекты – [Электронный ресурс]. – Режим доступа: <https://национальныепроекты.рф/projects/tsifrovaya-ekonomika> (Дата обращения: 27.03.2024).

5. Федеральный проект «Искусственный интеллект»// Национальный проект «Цифровая экономика» // Национальные проекты – [Электронный ресурс]. – Режим доступа: <https://национальныепроекты.рф/projects/tsifrovaya-ekonomika/p-iskusstvennyu-intellekt-p> (Дата обращения: 27.03.2024).

Описание системы оценки качества освоения модуля 2.

Контроль результатов освоения модуля 2 осуществляется в ходе текущего контроля успеваемости и промежуточной аттестации.

Текущий контроль предусмотрен в ходе изучения каждой темы.

Формами текущего контроля являются тесты и выполнение практических работ.

Тесты содержат не менее 5 вопросов с одним или несколькими правильными ответами. За каждый правильный ответ ставится 1 балл. Критерием прохождения теста является получение не менее 75% правильных ответов.

Примеры тестов.

Модуль 2. Профильный.

Раздел 2.2. Построение архитектур больших данных.

Тема: Масштабирование и автомасштабирование архитектур больших данных.

Мониторинг в Kubernetes.

Вопросы:

1. Какое понятие используется для описания изменения масштаба приложений в Kubernetes на основе нагрузки?

- a) Vertical Pod Autoscaler
 - b) Horizontal Pod Autoscaler (HPA)**
 - c) Pod Resizer
 - d) Cluster Autoscaler
2. Какие инструменты используются для сбора и визуализации метрик в Kubernetes?
- a) Jenkins и Docker
 - b) Kubernetes Dashboard и Kibana
 - c) Prometheus и Grafana**
 - d) AWS CloudWatch и Azure Monitor

3. Какой из следующих инструментов используется для мониторинга состояния кластера в Kubernetes?

- a) Jenkins
- b) Apach
- c) Prometheus**
- d) Elasticsearch

4. Что означает HPA с настройкой "targetCPUUtilizationPercentage: 80%"?

a) HPA будет увеличивать количество реплик до тех пор, пока загрузка CPU не достигнет 80%

b) HPA будет уменьшать количество реплик до тех пор, пока загрузка CPU не достигнет 80%

c) HPA будет поддерживать количество реплик на уровне, при котором загрузка CPU не превышает 80%

d) HPA не имеет никакого отношения к загрузке CPU

5. Какие типы метрик обычно используются для настройки HPA?

- a) CPU и память**
- b) Диск и сеть
- c) Время ответа и количество запросов
- d) IP-адреса и домены.

Выполнение практических работ оценивается в бинарной системе: зачтено / не зачтено.

Зачтено: задача решена, могут быть недочеты и неточности в решении.

Не зачтено: задача не решена.

Примеры практических работ.

Модуль 2. Профильный.

Раздел 2.2. Построение архитектур больших данных.

Тема 9. NLP – обработка естественного языка.

Задача:

Использовать предоставленный датасет текстов для определения токсичности текстов.

1. Предобработать данные – разделить данные, очистить.
2. Векторизировать текст.
3. Выбрать любую модель машинного обучения и обучить на данных.
4. Рассмотреть метрики качества, вывести результат.

Ссылку на репозиторий с кодом прислать в чат с преподавателем.

Промежуточная аттестация проводится в формате решения практико-ориентированной задачи (кейса).

Примеры практико-ориентированных задач (кейсов).

Модуль 2.

Задание 1: Разработка и обучение модели машинного обучения с использованием PySpark.

Цель: Освоить процесс разработки и обучения модели машинного обучения на больших данных с использованием PySpark.

Шаги:

1. Создайте кластер Spark с помощью PySpark.
2. Загрузите набор данных, подходящий для задачи классификации или регрессии.
3. Проведите предварительный анализ данных, включая обработку пропущенных значений и кодирование категориальных признаков.
4. Разделите данные на обучающую и тестовую выборки.
5. Используйте PySpark для обучения модели машинного обучения на обучающих данных.
6. Оцените качество модели на тестовой выборке с использованием соответствующих метрик.
7. Попробуйте улучшить качество модели, изменяя гиперпараметры или применяя другие алгоритмы машинного обучения.
8. Составьте отчет о результатах обучения модели, включая описание выбранной модели, ее характеристики и процесс оптимизации.

Ссылку на составленный отчет прислать в чат с преподавателем.

Задание 2: Развертывание и управление ML-моделью в Kubernetes с Kubeflow.

Цель: Научиться разворачивать и управлять ML-моделью в среде Kubernetes с использованием Kubeflow.

Шаги:

1. Установите и настройте Minikube для локальной разработки и тестирования Kubernetes кластера.
2. Установите Kubeflow на ваш Kubernetes кластер.

3. Подготовьте ML-модель, которую вы хотите развернуть, и ее зависимости.
4. Создайте контейнер для вашей ML-модели с помощью Docker и опубликуйте его в Docker Hub.
5. Создайте манифесты Kubernetes для развертывания ML-модели, включая сервис и деплоймент.
6. Разверните ML-модель в вашем Kubernetes кластере с помощью Kubeflow.
7. Настройте мониторинг и логирование вашей ML-модели с использованием инструментов, предоставляемых Kubeflow.
8. Оптимизируйте и масштабируйте ML-модель по мере необходимости.
9. Проведите тестирование развернутой ML-модели, убедившись, что она корректно работает в Kubernetes среде.
10. Составьте документацию по развертыванию и управлению ML-моделью с использованием Kubeflow.

Ссылку на составленную документацию прислать в чат с преподавателем.

Критерии и шкала оценивания:

0-4 балла: имеются содержательные и логические ошибки, решение кейса не найдено.
5-6 баллов: решение кейса в целом найдено, но оно неоптимально и/или имеются логические ошибки.

7-8 баллов: решение кейса найдено, но имеются неточности в решении.

9-10 баллов: решение кейса найдено, ошибки отсутствуют.

Максимально возможное число баллов за работу – 10.

не менее 9 баллов – «отлично».

7-8 баллов – «хорошо».

5-6 баллов – «удовлетворительно».

0-4 балла – «неудовлетворительно».

5. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ.

Специфика организационных действий и педагогических условий.

Для достижения планируемых результатов обучение строится с использованием следующих:

методов: case-study, метод проектов, модульное обучение, проблемное обучение, контекстное обучение;

форм: лекции с использованием мультимедиа, практические занятия, самостоятельная работа.

Кроме того, обучение строится с применением **технологий** электронного обучения и дистанционных образовательных технологий.

Кадровое обеспечение программы (преподавательский состав).

К реализации программы привлечены представители образовательных организаций высшего образования и представители компаний со стажем работы в области искусственного интеллекта и в смежных областях.

Представители образовательных организаций высшего образования имеют высшее образование, ученую степень кандидата или доктора наук, стаж научно-педагогической работы более трех лет, а также не выполняют функции иностранного агента.

Представители компаний со стажем работы в области искусственного интеллекта и в смежных областях имеют опыт решения практических задач с использованием технологий искусственного интеллекта более 3 лет в течение последних 10 лет в профильной компании или в профильном подразделении, а также не выполняют функции иностранного агента.

Члены преподавательского состава имеет за последние 3 года научные публикации, соответствующие направлению данной программы, в журналах, включенных в перечень ВАК (К1 и К2), а также в журналах, включенных в «Белый список» Минобрнауки РФ 1-го квартиля.

№ п/п	Фамилия, имя, отчество (при наличии)	Место основной работы и должность, ученая степень и ученое звание (при наличии)	Ссылки на веб-страницы с портфолио (при наличии)
1.	Борисов Вадим Владимирович <i>«Белый список» – 4 шт, из них 1-ого квартиля – 1 шт. ВАК К1 – 2 шт.</i>	профессор кафедры вычислительной техники филиала НИУ «МЭИ» в г. Смоленске, д.т.н., профессор	

2.	Мещеряков Роман Валерьевич <i>«Белый список» – 1 шт. ВАК К1 – 3 шт. ВАК К2 – 6 шт.</i>	главный научный сотрудник ИПУ РАН, д.т.н., профессор	
3.	Панкратов Денис Юрьевич <i>«Белый список» – 1 шт. ВАК К1 – 6 шт. ВАК К2 – 1 шт.</i>	доцент кафедры «Системы и сети радиосвязи и телерадиовещания» факультета Радио и телевидение МТУСИ, к.т.н.	
4.	Запорожец Марина Владимировна	инженер данных ООО «1Т»	
5.	Набока Михаил Викторович	ведущий BI Data Analyst, индивидуальный предприниматель, к.т.н.	https://data.1t.ru/naboka
6.	Никаноров Иван Михайлович <i>ВАК К1 – 1 шт.</i>	ведущий разработчик ООО «1Т»	https://data.1t.ru/nikanorov
7.	Семенов Анатолий Сергеевич <i>ВАК К1 – 1 шт.</i>	старший преподаватель по ИИ, ООО «1Т»	https://data.1t.ru/semenenko
8.	Семчук Дмитрий Борисович	ведущий системный аналитик ООО «1Т»	
9.	Шумаков Максим Витальевич <i>ВАК К1 – 1 шт.</i>	ведущий разработчик ООО «1Т»	
10.	Трефилов Павел Александрович	разработчик ООО «1Т»	https://data.1t.ru/trefilov
11.	Абанкина Ирина Всеволодовна <i>«Белый список» – 1 шт. ВАК К1 – 3 шт.</i>	профессор, главный научный сотрудник Центра финансово-экономических решений в образовании НИУ «Высшая школа экономики», к.э.н.	
12.	Борисов Иван Сергеевич	системный аналитик ООО «1Т»	

Материально-технические условия реализации программы.

Вид занятий	Наименование оборудования, программного обеспечения
Лекционные занятия	Персональный компьютер с установленным на нем: Windows 10-11, x64/x86; от 8 Gb RAM; от 128 Gb SSD/HDD, монитор от 15"; сетевой интерфейс Fast Ethernet 100 Мбит; веб-браузеры Google Chrome, Mozilla Firefox, Opera, Microsoft Edge, Яндекс.Браузер и др.
Практические занятия, самостоятельная работа, промежуточная и итоговая аттестация	Персональный компьютер с установленным на нем: Windows 10 и выше, x64/x86; от 8 Gb RAM; от 128 Gb SSD/HDD, монитор от 15"; сетевой интерфейс Fast Ethernet 100 Мбит; веб-браузеры Google Chrome, Mozilla Firefox, Opera, Microsoft Edge, Яндекс.Браузер и др. Anaconda 2.7 или 3.5 Доступ к облачным вычислительным ресурсам

Материально-технические условия соответствуют действующим санитарным и противопожарным правилам и нормам.

При проведении учебных занятий с применением дистанционных образовательных технологий (ДОТ) у слушателя должен быть персональный компьютер, оснащенный аудиокolonками, с доступом в сеть интернет и установленным видеоплеером, способным воспроизводить видеофайлы.

Выдаваемый документ при успешном освоении программы.

Удостоверение о повышении квалификации ООО «IT»

6. СИСТЕМА ОЦЕНКИ КАЧЕСТВА ОСВОЕНИЯ ПРОГРАММЫ.

В систему оценки качества освоения программы входят:

- 1) текущий контроль;
- 2) промежуточная аттестация;
- 3) итоговая аттестация.

Формы, методы проведения и оценочные материалы текущего контроля и промежуточной аттестации представлены в соответствующих рабочих программах модулей.

Для зачисления на курс потенциальному слушателю необходимо пройти входную диагностику (вступительное испытание).

Входная диагностика (вступительное испытание).

Входная диагностика (вступительное испытание) проводится в тестовой форме.

1. Какой командой можно вывести текущий каталог в командной строке?
 - a. dir
 - b. pwd**
 - c. ls
 - d. cd
2. Какой из нижеприведенных терминов является аббревиатурой для "Structured Query Language"?
 - a. JSON
 - b. SQL**
 - c. XML
 - d. CSV
3. Что такое "git" в контексте разработки программного обеспечения?
 - a. Язык программирования
 - b. Система управления базами данных
 - c. Протокол передачи данных
 - d. Система контроля версий**
4. Что такое "плоскость рассеяния" в контексте анализа данных?
 - a. Тип данных
 - b. Метод обработки ошибок
 - c. Визуализация взаимосвязи двух переменных**
 - d. Функция распределения

5. Какая из перечисленных ниже задач является типичной задачей машинного обучения "обучение с учителем"?

- a. Кластеризация
- b. Прогнозирование временных рядов

c. Классификация

d. Уменьшение размерности данных

6. Что такое "нормализация" данных в контексте анализа данных?

a. Процесс преобразования данных в формат JSON

b. Метод преобразования данных к определенному диапазону

c. Процедура удаления выбросов из данных

d. Форматирование данных в структурированный вид

7. Какая из перечисленных ниже структур данных обычно используется для хранения неупорядоченных и уникальных элементов?

a. Стек

b. Очередь

c. Множество

d. Список

8. Какая функция используется для чтения данных из файла в Python?

a. read_file()

b. open_file()

c. read()

d. load_data()

9. Какой метод статистического анализа используется для определения связи между двумя переменными?

a. Корреляция

b. Стандартизация

c. Дисперсия

d. Ковариация

10. Что такое "нейронная сеть" в контексте машинного обучения?

a. Способ представления данных в виде графа

b. Метод анализа больших данных

c. Модель, инспирированная работой человеческого мозга

d. Формат хранения изображений

11. Что такое "гипотеза" в контексте статистики и анализа данных?

a. Предположение о значении параметра популяции

- b. Метод сбора данных
- c. Математическая формула
- d. Показатель качества модели

12. Какой метод используется для обучения модели машинного обучения на непомеченных данных?

- a. Обучение с учителем
- b. Обучение без учителя**
- c. Подкрепленное обучение
- d. Передача обучения

13. Что такое "рекурсия" в программировании?

- a. Процесс обучения модели поэтапно
- b. Применение функции к самой себе**
- c. Подход к анализу данных
- d. Определение типа переменной

14. Что такое «кластеризация» в контексте машинного обучения?

- a. Процесс разделения данных на группы на основе их характеристик**
- b. Метод предсказания временных рядов
- c. Функция активации нейронной сети
- d. Метод анализа больших данных

15. Что такое «регуляризация» в контексте машинного обучения?

- a. Метод для уменьшения шума в данных
- b. Процедура оценки качества модели
- c. Метод добавления штрафа за сложность модели**
- d. Способ визуализации данных

16. Что такое «матрица ошибок» в контексте машинного обучения?

- a. Метод предсказания временных рядов
- b. Метод обработки пропущенных данных
- c. Таблица, используемая для оценки качества модели**
- d. Функция активации нейронной сети

17. Какой тип алгоритмов машинного обучения используется для задач классификации?

- a. Кластеризация
- b. Регрессия

c. Деревья решений

d. SVM

18. Что такое «бутстрап» в контексте статистики и анализа данных?

a. Метод оценки доверительных интервалов

b. Метод предсказания временных рядов

c. Метод визуализации данных

d. Функция активации нейронной сети

19. Какой метод используется для заполнения пропущенных значений в данных?

a. Удаление строк с пропущенными значениями

b. Заполнение средним значением

c. Интерполяция

d. Все вышеперечисленное

20. Что такое «условная вероятность» в контексте теории вероятностей?

a. Вероятность события при условии, что другое событие уже произошло

b. Вероятность события, происходящего при любых обстоятельствах

c. Вероятность события, происходящего сразу после другого события

d. Вероятность события, происходящего при всех возможных обстоятельствах

21. Что такое «перцептрон» в контексте машинного обучения?

a. Метод обработки изображений

b. Алгоритм кластерного анализа

c. Простейшая форма искусственной нейронной сети

d. Метод кластеризации

22. Что такое «переобучение» в контексте машинного обучения?

a. Процесс обучения модели на большем объеме данных

b. Ситуация, когда модель слишком хорошо запоминает обучающие данные и плохо обобщает на новые данные

c. Ошибка в модели

d. Недостаточное количество данных для обучения модели

23. Какой тип алгоритмов машинного обучения используется для задач прогнозирования?

a. Классификация

b. Кластеризация

c. Регрессия

d. Деревья принятия решений

24. Что такое «статистическая значимость» в контексте анализа данных?

a. Уровень вероятности, при котором отвергается нулевая гипотеза

b. Мера точности модели

c. Степень зависимости двух переменных

d. Мера разброса данных

25. Что такое «векторизация» в контексте анализа данных?

a. Метод обработки текстовых данных

b. Преобразование данных в векторный формат

c. Метод кластеризации данных

d. Применение функции к каждому элементу массива данных

Пороговое значение для успешного прохождения вступительного испытания (не менее 65% от общего количества результатов выполнения заданий).

Итоговая аттестация.

Итоговая аттестация проходит в форме решения практико-ориентированных задач (кейсов).

Практико-ориентированные задачи (кейсы) для итоговой аттестации предоставлены организациями, работающими в области искусственного интеллекта и смежных областях (ООО «Альмира»), а также организациями, применяющими технологии интеллектуальной обработки данных для решения своих производственных задач (ПК «ИТ Союз», ООО «Верконт Сервис», ООО СП «Содружество», ООО ЭЦ «Социология и аналитика»).

Практико-ориентированная задача (кейс) 1. Прогнозирование временных рядов температуры.

Цель: Построить модель прогнозирования температуры на основе исторических данных и подготовить Docker контейнер для дальнейшего использования модели.

Условие:

1. Загрузите набор данных о погоде, содержащий информацию о температуре в различные временные периоды.

2. Проведите предварительный анализ данных: изучите структуру набора данных, проверьте наличие пропущенных значений, исследуйте временной ряд.

3. Разделите данные на обучающую и тестовую выборки.

4. Постройте модель прогнозирования температуры на основе выбранной архитектуры нейронной сети или другого подходящего метода машинного обучения.
5. Оцените качество модели на тестовой выборке с использованием подходящих метрик (например, RMSE, MAE).
6. Подготовьте документацию, описывающую шаги по подготовке данных, построению модели и оценке ее качества.
7. Разработайте Docker контейнер, включающий в себя модель прогнозирования температуры и необходимые зависимости.
8. Проверьте работоспособность Docker контейнера, запустив его локально.

Результаты:

1. Подготовлена документация, описывающая процесс работы с данными, построением модели и оценкой ее качества.
2. Разработан Docker контейнер с моделью прогнозирования температуры и необходимыми зависимостями.

Практико-ориентированная задача (кейс) 2. Анализ временных рядов для прогнозирования трафика в сети.

Цель: Разработать модель анализа временных рядов для прогнозирования трафика в компьютерной сети с целью оптимизации ресурсов и предотвращения возможных перегрузок.

Условие:

1. Получите данные о трафике в компьютерной сети за определенный период времени, включая объемы передачи данных, пиковые нагрузки, часы пиковой активности и т. д.
2. Используйте язык программирования Python и библиотеки для анализа временных рядов, такие как Pandas, NumPy, и Statsmodels, для предобработки данных и разработки модели прогнозирования.
3. Разработайте модель анализа временных рядов, например, ARIMA или SARIMA, для прогнозирования трафика в сети.
4. Проведите анализ точности прогнозов модели на основе сравнения прогнозных значений с фактическими данными.
5. Создайте документацию, описывающую процесс анализа данных, разработки модели и методологию прогнозирования трафика в сети.

6. Разработайте Docker контейнер, включающий модель анализа временных рядов и необходимые библиотеки.

7. Гарантируйте готовность Docker контейнера к развертыванию и использованию в реальных условиях сети.

Результаты:

1. Разработана модель анализа временных рядов для прогнозирования трафика в компьютерной сети.

2. Создан Docker контейнер, содержащий модель анализа временных рядов и необходимые библиотеки.

Практико-ориентированная задача (кейс) 3. Распознавание лиц на изображениях с использованием нейронных сетей.

Цель: Разработать модель для распознавания лиц на изображениях и создать Docker контейнер для развертывания модели.

Условие:

1. Соберите набор данных изображений лиц различных людей для обучения модели.
2. Проведите предварительную обработку данных: масштабирование, выравнивание и т.д.

3. Разделите данные на обучающую и тестовую выборки.
4. Разработайте нейронную сеть или используйте предобученную модель для распознавания лиц.

5. Обучите модель на обучающей выборке и оцените ее производительность на тестовой выборке.

6. Подготовьте документацию, описывающую процесс сбора данных, подготовки данных, обучения модели и ее оценки.

7. Создайте Docker контейнер, который включает в себя модель для распознавания лиц и необходимые библиотеки.

8. Убедитесь, что Docker контейнер корректно разворачивается и модель работает в нем.

Результаты:

1. Подготовлена документация, описывающая процесс сбора данных, подготовки данных, обучения модели и оценки ее производительности.

2. Создан Docker контейнер, содержащий модель для распознавания лиц и необходимые библиотеки.

Практико-ориентированная задача (кейс) 4. Анализ данных транспортных потоков с использованием машинного обучения и контейнеризации.

Цель: Провести анализ данных транспортных потоков для оптимизации инфраструктуры и создать Docker контейнер для развертывания аналитической модели.

Условие:

1. Получите данные о транспортных потоках, включающие информацию о движении транспортных средств, времени их движения, скорости и других параметрах.
2. Проведите предварительный анализ данных и выделите основные тренды и закономерности в транспортных потоках.
3. Разработайте модель машинного обучения для прогнозирования и анализа транспортных потоков на основе полученных данных.
4. Обучите модель на исторических данных о транспортных потоках.
5. Оцените производительность модели на тестовых данных и ее применимость для оптимизации транспортной инфраструктуры.
6. Подготовьте документацию, описывающую процесс сбора данных, предварительного анализа, разработки модели и оценки ее эффективности.
7. Создайте Docker контейнер, включающий модель анализа данных транспортных потоков и необходимые библиотеки.
8. Убедитесь, что Docker контейнер корректно развертывается и модель работает в нем.

Результаты:

1. Подготовлена документация, описывающая процесс сбора данных, предварительного анализа, разработки модели и оценки ее эффективности.
2. Создан Docker контейнер, содержащий модель анализа данных транспортных потоков и необходимые библиотеки.

Практико-ориентированная задача (кейс) 5. Автоматизация процесса обработки текстовых данных с использованием Python и Docker.

Цель: Разработать скрипт для автоматизации обработки текстовых данных и создать Docker контейнер для его развертывания.

Условие:

1. Получите набор текстовых данных для обработки. Это могут быть, например, отзывы пользователей, новостные статьи или техническая документация.

2. Разработайте скрипт на языке Python для обработки текстовых данных. Скрипт должен включать в себя следующие этапы обработки:

- Загрузка текстовых данных из источника.
- Предварительная обработка текста (токенизация, удаление стоп-слов, лемматизация и т. д.).
- Применение алгоритмов анализа текста (например, определение тональности, категоризация текста и т. д.).
- Генерация отчета или другого вывода на основе анализа текста.

3. Подготовьте документацию, описывающую структуру и функциональность разработанного скрипта.

4. Создайте Docker контейнер, включающий разработанный скрипт и все необходимые зависимости.

5. Убедитесь, что Docker контейнер корректно разворачивается и скрипт успешно выполняется.

Результаты:

1. Разработан скрипт на языке Python для обработки текстовых данных.
2. Подготовлена документация, описывающая структуру и функциональность разработанного скрипта.
3. Создан Docker контейнер, включающий разработанный скрипт и все необходимые зависимости.

Практико-ориентированная задача (кейс) 6. Разработка модели машинного обучения для предсказания спроса на товары в розничной торговле.

Цель: Создать модель машинного обучения для прогнозирования спроса на товары в розничной торговле.

Условие:

1. Получите набор данных о продажах товаров в розничной торговле. Набор данных должен включать в себя информацию о товарах, времени продажи, количестве проданных единиц и другие связанные параметры.

2. Проведите предварительный анализ данных, включающий в себя исследование распределения спроса на товары во времени, выявление сезонных и трендовых паттернов, а также анализ влияния различных факторов на спрос.

3. Разработайте модель машинного обучения для предсказания спроса на товары. Выберите подходящий алгоритм обучения, проведите обучение модели на исторических данных и оцените ее качество.

4. Протестируйте работу модели на новых данных, чтобы оценить ее точность и надежность прогнозов.

5. Создайте документацию, описывающую структуру и функциональность разработанной модели.

6. Разработайте Docker контейнер, включающий в себя модель машинного обучения и необходимые зависимости.

7. Убедитесь, что Docker контейнер корректно разворачивается и модель успешно выполняет прогнозирование спроса на товары.

Результаты:

1. Разработана модель машинного обучения для предсказания спроса на товары в розничной торговле.

2. Подготовлена документация, описывающая структуру и функциональность разработанной модели.

3. Создан Docker контейнер, включающий в себя модель машинного обучения и необходимые зависимости.

Практико-ориентированная задача (кейс) 7. Оптимизация производственных процессов с использованием алгоритмов машинного обучения.

Цель: Разработать модель машинного обучения для оптимизации производственных процессов на предприятии.

Условие:

1. Получите данные о производственных процессах на предприятии, включая информацию о времени, затраченном на выполнение различных операций, параметрах оборудования, технических характеристиках и другие соответствующие параметры.

2. Проведите анализ данных, выявив основные факторы, влияющие на эффективность производственных процессов, и определите области, где возможно повышение производительности.

3. Разработайте модель машинного обучения для прогнозирования времени выполнения производственных задач, оптимизируйте ее на основе исторических данных и выберите наилучший алгоритм для данной задачи.

4. Протестируйте работу модели на новых данных и оцените ее точность и эффективность.

5. Создайте документацию, описывающую процесс разработки модели и принцип ее работы.

6. Разработайте Docker контейнер, содержащий модель машинного обучения и необходимые библиотеки.

7. Убедитесь, что Docker контейнер успешно запускается и модель готова к использованию в производственной среде.

Результаты:

1. Разработана модель машинного обучения для оптимизации производственных процессов на предприятии.

2. Подготовлена документация, описывающая процесс разработки модели и ее функциональность.

3. Создан Docker контейнер, содержащий модель машинного обучения и необходимые библиотеки.

Практико-ориентированная задача (кейс) 8. Анализ пользовательского поведения в приложении для оптимизации интерфейса.

Цель: Провести анализ пользовательского поведения в мобильном приложении и оптимизировать его интерфейс для улучшения пользовательского опыта.

Условие:

1. Получите данные о поведении пользователей в мобильном приложении, включая действия, совершенные пользователями, время нахождения в приложении, покупки и другие взаимодействия.

2. Проведите анализ данных, выявив наиболее часто встречающиеся паттерны поведения пользователей, и выделите основные проблемы в интерфейсе приложения.

3. Разработайте модель анализа пользовательского поведения с использованием методов машинного обучения и статистических подходов.

4. Проанализируйте результаты модели и определите области для улучшения интерфейса приложения, которые могут повысить его удобство и эффективность.

5. Внесите изменения в интерфейс приложения на основе полученных результатов.

6. Создайте документацию, описывающую процесс анализа пользовательского поведения и предложенные улучшения интерфейса.

7. Разработайте Docker контейнер, содержащий модель анализа пользовательского поведения и необходимые библиотеки.

8. Убедитесь, что Docker контейнер успешно запускается и модель готова к использованию для анализа данных в реальном времени.

Результаты:

1. Проведен анализ пользовательского поведения в мобильном приложении.
2. Внесены улучшения в интерфейс приложения на основе результатов анализа.
3. Создан Docker контейнер, содержащий модель анализа пользовательского поведения и необходимые библиотеки.

Практико-ориентированная задача (кейс) 9. Создание системы рекомендаций для электронной коммерции.

Цель: Разработать систему рекомендаций для интернет-магазина с целью увеличения конверсии и улучшения пользовательского опыта.

Условие:

1. Соберите данные о предыдущих покупках пользователей, просмотрах товаров, оценках и других взаимодействиях с платформой интернет-магазина.
2. Используйте язык программирования Python и библиотеки машинного обучения, такие как scikit-learn и pandas, для анализа данных и разработки модели рекомендаций.
3. Разработайте модель рекомендаций, которая будет учитывать предпочтения пользователей, их историю покупок и схожие интересы с другими пользователями.
4. Протестируйте работу модели на новых данных и оцените ее эффективность с помощью метрик, таких как точность и полнота рекомендаций.
5. Создайте документацию, описывающую процесс анализа данных, разработки модели и принципы работы системы рекомендаций.
6. Разработайте Docker контейнер, содержащий модель системы рекомендаций и необходимые библиотеки.
7. Убедитесь, что Docker контейнер готов к развертыванию и использованию в реальных условиях интернет-магазина.

Результаты:

1. Разработана система рекомендаций для интернет-магазина на основе анализа пользовательского поведения.
2. Создан Docker контейнер, содержащий модель системы рекомендаций и необходимые библиотеки.

Практико-ориентированная задача (кейс) 10. Анализ пользовательского поведения в социальных сетях.

Цель: Провести анализ поведения пользователей в социальных сетях с использованием методов машинного обучения.

Условие:

1. Получите данные о пользовательской активности в социальной сети, включая лайки, комментарии, подписки, и т. д.
2. Используйте язык программирования Python и библиотеки для анализа данных, такие как Pandas, NumPy, и Matplotlib, для предобработки и визуализации данных.
3. Проанализируйте пользовательские взаимодействия, выявите популярные темы, предпочтения пользователей, идентифицируйте ключевых пользователей и т. д.
4. Примените методы машинного обучения, например, кластеризацию или классификацию, для сегментации пользователей или прогнозирования их дальнейших действий.
5. Оцените результаты анализа и эффективность примененных методов.
6. Создайте документацию, описывающую процесс анализа данных, используемые методы и полученные результаты.
7. Разработайте Docker контейнер, содержащий все необходимые компоненты для проведения анализа пользовательского поведения в социальных сетях.

Результаты:

1. Проведен анализ пользовательского поведения в социальных сетях.
2. Разработан Docker контейнер для проведения анализа данных.
3. Создана документация, описывающая процесс анализа данных, используемые методы и полученные результаты.

Практико-ориентированная задача (кейс) 11. Оптимизация энергопотребления в зданиях.

Цель: Разработать модель для оптимизации энергопотребления в зданиях с использованием анализа данных.

Условие:

1. Получите данные о потреблении энергии в зданиях, включая данные о температуре, влажности, освещенности, активности людей и т. д.

2. Проанализируйте данные, выявите связи между параметрами и энергопотреблением.

3. Используйте методы машинного обучения для разработки модели, которая может оптимизировать энергопотребление в зданиях.

4. Учтите в модели сезонные и временные изменения, а также особенности каждого конкретного здания.

5. Протестируйте модель на новых данных и оцените ее эффективность.

6. Разработайте документацию, объясняющую принцип работы модели, ее преимущества и способы использования.

7. Создайте Docker контейнер, включающий модель и необходимые библиотеки для ее работы.

Результаты:

1. Разработана модель для оптимизации энергопотребления в зданиях.

2. Создан Docker контейнер с моделью и необходимыми библиотеками.

3. Подготовлена документация, объясняющая принцип работы модели и способы ее использования.

Практико-ориентированная задача (кейс) 12. Анализ пользовательского поведения в онлайн-магазине.

Цель: Разработать модель для анализа пользовательского поведения в онлайн-магазине с использованием Python.

Условие:

1. Получите данные о действиях пользователей в онлайн-магазине, включая просмотры товаров, добавление товаров в корзину, совершение покупок и т. д.

2. Используя библиотеки Python (Pandas, NumPy, Matplotlib), проведите анализ данных и выявите основные паттерны пользовательского поведения.

3. Разработайте модель для прогнозирования совершения покупок пользователями на основе их предыдущего поведения.

4. Протестируйте модель на новых данных и оцените ее точность и эффективность.

5. Напишите скрипт на Python, который позволит в реальном времени анализировать действия пользователей и давать рекомендации по улучшению их опыта.

6. Разработайте документацию, объясняющую основные шаги анализа данных, построения модели и использования скрипта.

7. Создайте Docker контейнер, включающий все необходимые компоненты для работы анализа пользовательского поведения.

Результаты:

1. Разработана модель для анализа пользовательского поведения в онлайн-магазине.
2. Написан скрипт на Python для анализа действий пользователей в реальном времени.
3. Создан Docker контейнер с моделью, скриптом и необходимыми библиотеками.
4. Подготовлена документация, объясняющая основные шаги анализа данных и использования модели.

Практико-ориентированная задача (кейс) 13. Оптимизация производительности веб-приложения.

Цель: Разработать стратегии оптимизации производительности веб-приложения для повышения отзывчивости и эффективности работы.

Условие:

1. Анализируя текущее состояние веб-приложения, выявите узкие места и проблемные зоны, влияющие на его производительность, с использованием инструментов Python.
2. Используя инструменты мониторинга производительности и анализа запросов на Python, соберите данные о времени отклика, загрузке ресурсов и прочих метриках производительности.
3. Разработайте стратегии оптимизации, включая кэширование данных, оптимизацию запросов к базе данных на Python, улучшение алгоритмов обработки данных и т. д.
4. Создайте Docker-контейнер для веб-приложения, чтобы обеспечить изолированное и надежное развертывание.
5. Протестируйте оптимизированное веб-приложение на предмет улучшения производительности и сравните результаты с изначальным состоянием.
6. Напишите отчет, в котором подробно описаны проблемы производительности, предложенные стратегии оптимизации и результаты тестирования.
7. Разработайте документацию, содержащую рекомендации по оптимизации производительности для будущих проектов и поддержки веб-приложения.

Результаты:

1. Проведен анализ производительности веб-приложения на Python и выявлены узкие места.
2. Разработаны стратегии оптимизации на Python и создан Docker-контейнер для веб-приложения.
3. Проведено тестирование оптимизированного веб-приложения и сравнение результатов с изначальным состоянием.
4. Создан отчет о проблемах производительности, предложенных стратегиях оптимизации, результатам тестирования и документация с рекомендациями по оптимизации производительности.

Практико-ориентированная задача (кейс) 14. Управление зависимостями в веб-приложении.

Цель: Разработать стратегии управления зависимостями в веб-приложении для обеспечения стабильной работы и удобства развертывания.

Условие:

1. Изучите текущую конфигурацию зависимостей в веб-приложении, проведя анализ используемых библиотек и пакетов на Python.
2. Определите проблемные и устаревшие зависимости, а также потенциальные конфликты между версиями пакетов.
3. Разработайте стратегию управления зависимостями, включая частоту обновлений, использование виртуальных окружений и автоматизацию процесса управления зависимостями.
4. Проанализируйте влияние изменений в зависимостях на производительность и стабильность работы веб-приложения.
5. Реализуйте предложенные стратегии управления зависимостями и протестируйте их на практике.
6. Внедрите докеризацию веб-приложения для обеспечения удобства развертывания и изоляции окружения.
7. Создайте документацию, описывающую разработанную стратегию управления зависимостями и процесс докеризации.
8. Подготовьте отчет, содержащий результаты анализа зависимостей, разработанную стратегию и описание процесса докеризации.

Результаты:

1. Разработана стратегия управления зависимостями, включая частоту обновлений и использование виртуальных окружений.
2. Реализованы предложенные стратегии и протестированы на практике.
3. Внедрена докеризация веб-приложения для облегчения развертывания и изоляции окружения, включая создание Docker-контейнера.
4. Создана документация, описывающая процесс управления зависимостями и докеризации.

Практико-ориентированная задача (кейс) 15. Интеграция Apache Kafka для обработки данных в системе аналитики.

Цель: Разработать процесс интеграции Apache Kafka для обработки и передачи данных в системе аналитики.

Условие:

1. Настройте Apache Kafka в среде разработки с использованием Docker.
2. Создайте тему Kafka для передачи данных между различными компонентами системы.
3. Разработайте Python скрипт для генерации тестовых данных и их отправки в тему Kafka.
4. Напишите скрипт для чтения данных из темы Kafka и сохранения их в хранилище данных (например, файловая система или база данных).
5. Используйте Apache Kafka Connect для интеграции Kafka с хранилищем данных.

Результат:

1. Документация, описывающая процесс настройки и использования Apache Kafka в среде разработки.
2. Docker-контейнер, включающий настроенный Apache Kafka и все необходимые компоненты для интеграции.
3. Скрипты на Python для генерации данных, их отправки в Kafka, чтения из Kafka и сохранения в хранилище данных.

Практико-ориентированная задача (кейс) 16. Разработка системы анализа данных с использованием Apache Spark.

Цель: Построить систему анализа данных на базе Apache Spark для обработки больших объемов информации.

Условие:

1. Настройте кластер Apache Spark с использованием Docker.
2. Разработайте процесс загрузки данных из внешних источников в кластер Apache Spark.
3. Произведите анализ данных с использованием Apache Spark SQL, DataFrame API и MLlib.
4. Реализуйте процесс сохранения результатов анализа данных в различных форматах (например, CSV, Parquet).
5. Оптимизируйте процесс обработки данных, используя возможности распределенных вычислений Apache Spark.

Результат:

1. Документация, описывающая настройку кластера Apache Spark и процессы обработки данных.
2. Docker-контейнер, содержащий настроенный кластер Apache Spark и необходимые инструменты.
3. Программный код на Scala или Python, реализующий процессы загрузки, анализа и сохранения данных с использованием Apache Spark.

Практико-ориентированная задача (кейс) 17. Разработка системы обработки и анализа потоков данных с использованием Apache Flink.

Цель: Создать систему для обработки и анализа потоков данных в режиме реального времени с помощью Apache Flink.

Условие:

1. Настройте кластер Apache Flink с использованием Docker.
2. Разработайте процесс обработки потоков данных из различных источников с помощью Apache Flink.
3. Реализуйте алгоритмы анализа потоков данных, такие как оконные функции, фильтрация, агрегация и т. д.
4. Используйте Apache Flink для обнаружения и обработки аномалий в потоках данных.
5. Произведите интеграцию с внешними системами хранения данных, такими как Apache Kafka, Apache Hadoop и другими.

Результат:

1. Документация, описывающая настройку кластера Apache Flink и реализованные процессы обработки и анализа данных.

2. Docker-контейнер, содержащий настроенный кластер Apache Flink и необходимые компоненты.

3. Программный код на Java или Scala, реализующий процессы обработки и анализа потоков данных с использованием Apache Flink.

Практико-ориентированная задача (кейс) 18. Разработка системы мониторинга и управления контейнеризированными приложениями с использованием Kubernetes.

Цель: Создать систему для мониторинга, управления и автомасштабирования контейнеризированными приложениями с помощью Kubernetes.

Условие:

1. Настройте кластер Kubernetes с использованием Minikube.
2. Разработайте механизм мониторинга состояния приложений в кластере Kubernetes с использованием Prometheus и Grafana.

3. Создайте автоматизированный процесс масштабирования приложений в зависимости от нагрузки с использованием Horizontal Pod Autoscaler.

4. Реализуйте механизм CI/CD для развертывания приложений в кластере Kubernetes с использованием GitLab CI.

5. Используйте Helm для управления пакетами приложений в Kubernetes.

Результат:

1. Документация, описывающая настройку кластера Kubernetes, механизм мониторинга и автомасштабирования, а также процесс CI/CD.

2. Docker-образы с настроенным кластером Kubernetes и компонентами мониторинга.

3. Автоматизированный процесс развертывания и масштабирования приложений в кластере Kubernetes.

Практико-ориентированная задача (кейс) 19. Разработка системы обработки и анализа данных с использованием Python и библиотек для машинного обучения.

Цель: Создать инструмент для обработки и анализа данных, а также для построения моделей машинного обучения на языке Python.

Условие:

1. Загрузка и предварительная обработка данных с использованием библиотеки Pandas.

2. Визуализация данных с помощью Matplotlib и Seaborn для выявления зависимостей и особенностей набора данных.

3. Применение методов машинного обучения с использованием библиотеки Scikit-learn для построения моделей классификации, регрессии или кластеризации в зависимости от характера данных.

4. Оценка качества моделей с использованием метрик соответствующего типа (например, Accuracy, F1-score, RMSE).

5. Разработка документации, описывающей процесс обработки данных, построения моделей и оценки их качества.

Результат:

1. Документация, включающая описание всех этапов обработки данных, построения моделей и их оценки, а также примеры кода.

2. Docker-контейнер с готовым окружением для работы с данными и запуска моделей машинного обучения.

Практико-ориентированная задача (кейс) 20. Разработка системы анализа и прогнозирования финансовых рынков с использованием Python и библиотек для финансового анализа.

Цель: Создать инструмент для анализа финансовых данных и прогнозирования изменений на финансовых рынках с использованием языка программирования Python и соответствующих библиотек.

Условие:

1. Сбор и загрузка финансовых данных с биржи, таких как котировки акций, валютные курсы и другие финансовые индексы, с использованием библиотеки Requests.

2. Предварительная обработка и анализ данных, включая очистку от выбросов, заполнение пропущенных значений и выделение основных финансовых показателей с использованием библиотеки Pandas.

3. Визуализация финансовых данных с помощью Matplotlib и Plotly для выявления трендов, сезонных изменений и других паттернов.

4. Применение методов статистического анализа для изучения корреляций между различными финансовыми индикаторами.

5. Разработка модели прогнозирования изменений на финансовых рынках на основе имеющихся данных с использованием библиотеки Scikit-learn или других библиотек машинного обучения.

6. Оценка качества прогнозов с использованием соответствующих метрик, таких как среднеквадратичная ошибка (RMSE) или коэффициент детерминации (R^2).

7. Подготовка документации, описывающей весь процесс анализа данных, построения модели и оценки ее качества в контексте финансовых рынков.

Результат:

1. Документация, содержащая описание шагов анализа данных, построения модели прогнозирования и оценки качества прогнозов в контексте финансовых рынков.

2. Docker-контейнер с необходимым программным обеспечением и зависимостями для работы с финансовыми данными и прогнозирования изменений на финансовых рынках.

Практико-ориентированная задача (кейс) 21. Разработка системы обнаружения аномалий в сетевом трафике с использованием Python и методов машинного обучения.

Цель: Создать инструмент для мониторинга сетевого трафика и обнаружения аномалий, указывающих на потенциальные кибератаки или нештатное поведение сетевых устройств.

Условие:

1. Загрузка и предварительная обработка данных о сетевом трафике, включая IP-адреса, порты, протоколы и объем переданных данных, с использованием библиотеки Pandas.

2. Визуализация характеристик сетевого трафика с помощью Matplotlib или других библиотек для выявления типичных паттернов и аномалий.

3. Применение методов машинного обучения, таких как кластеризация или классификация, для выявления аномального поведения в сетевом трафике.

4. Разработка модели обнаружения аномалий на основе имеющихся данных, используя библиотеки Scikit-learn или TensorFlow.

5. Оценка качества модели на тестовых данных с использованием соответствующих метрик, таких как точность, полнота и F1-мера.

6. Подготовка документации, описывающей весь процесс обработки и анализа сетевых данных, построения модели обнаружения аномалий и ее оценки.

Результат:

1. Документация, содержащая описание шагов обработки данных, построения модели обнаружения аномалий и оценки ее качества в контексте мониторинга сетевого трафика.

2. Docker-контейнер с необходимым программным обеспечением и зависимостями для работы с сетевыми данными и обнаружения аномалий в сетевом трафике.

Критерии оценивания, шкала оценивания.

0–4 балла: имеются содержательные и логические ошибки, решение кейса не найдено;

5–6 баллов: решение кейса в целом найдено, но оно неоптимально и/или имеются логические ошибки;

7–8 баллов: решение кейса найдено, но имеются неточности в решении;

9–10 баллов: решение кейса найдено, ошибки отсутствуют.

Максимально возможное число баллов – 10:

не менее 9 баллов – «отлично»;

7–8 баллов – «хорошо»;

5–6 баллов – «удовлетворительно»;

0–4 балла – «неудовлетворительно».

В ходе итоговой аттестации обучающиеся должны продемонстрировать следующие знания, умения и навыки:

Знать:

- функции, задачи, навыки, содержание работы архитектора данных;
- определения, историю развития и главные тренды больших данных и искусственного интеллекта;
- методы автоматизированной генерации и проверки гипотез в сфере исследовательской деятельности;
- основные принципы логического вывода и проверки гипотез в контексте анализа данных;
- теоретические основы алгоритмов, используемых для автоматизированного логического вывода и проверки гипотез;
- инструменты и технологии, применяемые для реализации автоматизированного логического вывода и проверки гипотез в сфере анализа данных;
- архитектуру, принципы, процесс, стадии и методологии разработки решений на основе ИИ;
- синтаксис и семантику основных языков искусственного интеллекта (NLP, UML, Python, JAVA/C#/C++/Scala) и основные приемы программирования на них;
- методы и инструментальные средства решения задач искусственного интеллекта;

- методы и программный инструментарий для тестирования программных компонентов;
- основные типы, источники больших данных, проблемы безопасности;
- методы и программный инструментарий технологий больших данных;
- базовые понятия и возможности теории игр для их применения в области машинного обучения и искусственного интеллекта;
- системы обработки и анализа больших массивов данных (SQL, NoSQL, Hadoop, SPARK и др.): принципы работы, функционал, язык запросов, преимущества и недостатки;
- технологии, методы и инструменты машинного обучения для решения профессиональных задач;
- нейронные сети: полносвязные, свёрточные и рекуррентные нейронные сети;
- методы обучения нейронных сетей;
- нейросетевые методы понижения размерности;
- архитектуры современных нейронных сетей и их использование для решения профессиональных задач
- методы, инструментальные средства и стандарты статистического и описательного анализа данных;
- основные платформы данных (облачные и внутрикорпоративные), применяемые при разработке решений на основе ИИ;
- основные виды представления данных: табличные, графовые, временные ряды;
- основные уровни представления данных;
- понятие сквозных цифровых субтехнологий искусственного интеллекта;
- методы и технологии внедрения сквозных цифровых субтехнологий искусственного интеллекта.

Уметь:

- осуществлять поиск информации о новых и перспективных методах анализа больших данных, выполнять сравнительный анализ методов;
- настраивать, конфигурировать и адаптировать программные средства автоматизированного логического вывода в сфере исследовательской деятельности;
- разрабатывать и настраивать алгоритмы автоматизированного логического вывода и проверки гипотез, учитывая специфику исследовательской деятельности;

- проектировать и настраивать конфигурации программных средств для эффективного применения методов автоматизированного логического вывода и проверки гипотез;
- проводить адаптацию и оптимизацию программных решений для учета требований и специфики конкретных задач;
- применять методы машинного обучения для тестирования программных компонентов;
- проводить тестирование систем искусственного интеллекта;
- применять методы машинного обучения для работы с большими данными;
- применять методы машинного обучения и нейронных сетей;
- поддерживать системы искусственного интеллекта на основе нейросетевых моделей и методов;
- использовать основные статистические методы анализа данных;
- использовать шины данных (kafka) при обработке больших данных;
- моделировать данные в хранилищах (DWH, Data Lake, NoSQL хранилища данных);
- внедрять технологии компьютерного зрения для обработки изображений и видео, задач биометрии;
- внедрять технологии обработки естественного языка для анализа и генерации текстов;
- внедрять технологии обучения с подкреплением для задач автоматизации производства и транспорта.

Владеть:

- языками программирования (Python, JAVA/C#/C++/Scala) для реализации методов анализа и структурирования данных;
- основными инструментами для работы с данными (Git, Docker, CI/CD, Jupyter notebook, kaggle, Nvidia Cuda, VS, IntelliJIdea);
- программными инструментами, позволяющими осуществлять автоматизированный логический вывод и проверку гипотез, такими как инструменты теории машинного обучения, статистические пакеты, инструменты символьного вычисления и другие соответствующие технологии;
- техниками и методами, используемыми для применения программных средств автоматизированного логического вывода в различных областях исследовательской деятельности;

- навыками использования различных инструментов и технологий в рамках процесса автоматизации логического вывода и проверки гипотез;
- навыками работы с основными библиотеками Python для визуализации данных (Matplotlib, Seaborn, Plotly);
- навыком создания логической и физической базы данных;
- навыком установки Hadoop на виртуальную машину;
- навыком использования Docker;
- навыком работы с MapReduce, PySpark;
- навыком BI-аналитики;
- навыком работы с NumPy;
- культурой постановки и планирования последовательности решения задач анализа данных и классификации;
- навыками анализа реальных задач из различных предметных областей на уровне отдельных подходов и коллективами алгоритмов;
- навыками работы с библиотеками, программными платформами для автоматизации процессов тестирования;
- навыками работы с библиотеками, программными платформами (фреймворками) и программными комплексами машинного обучения;
- навыком построения полносвязной нейронной сети для задачи классификации;
- основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных;
- способами улучшить качество модели с помощью методов Feature engineering;
- навыком использования библиотеки PyTorch, применения GPU в PyTorch;
- навыком построения полносвязной нейронной сети для задачи классификации;
- основными принципами, возможностями и порядком применения методов обучения с учителем и без учителя для решения задач классификации данных;
- методами машинного обучения и нейронных сетей;
- навыком создания логической и физической базы данных;
- навыком BI-аналитики;
- языками запросов для обращения к СУБД;
- навыком работы с генеративными моделями GAN для задач генерации;
- навыком работы с автокодировщиками для задач уменьшения размерности и поиска аномалий;
- навыком работы с технологией RL - обучения с подкреплением.

Тугой И.А.

Академический директор ООО «1Т»



Разработчики программы:

Борисов Вадим
Владимирович

Профессор кафедры вычислительной техники
филиала НИУ «МЭИ» в г. Смоленске, д.т.н.,
профессор



Мещеряков Роман
Валерьевич

Главный научный сотрудник ИПУ РАН, д.т.н.,
профессор



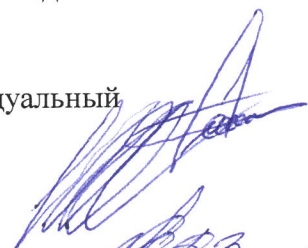
Панкратов Денис Юрьевич

Доцент кафедры «Системы и сети радиосвязи
и телерадиовещания» факультета «Радио и
телевидение» МТУСИ, к.т.н.



Набока Михаил Викторович

Ведущий BI Data Analyst, индивидуальный
предприниматель, к.т.н.



Борисов Иван Сергеевич

Системный аналитик ООО «1Т»

Запорожец Марина

Инженер данных ООО «1Т»



Владимировна

Никаноров Иван

Ведущий разработчик ООО «1Т»



Михайлович

Семененко Анатолий

Старший преподаватель по ИИ, ООО «1Т»



Сергеевич

Семчук Дмитрий Борисович

Системный аналитик ООО «1Т»



Шумаков Максим

Ведущий разработчик ООО «1Т»



Витальевич

Абанкина Ирина

Профессор, главный научный сотрудник



Всеволодовна

Центра финансово-экономических решений в
образовании НИУ «Высшая школа
экономики», к.э.н.



Трефилов Павел

Разработчик ООО «1Т»

Александрович