МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Колледж информационных и креативных технологий

РАБОЧАЯ ПРОГРАММА

МДК.01.05 Большие данные-платформы для поддержки ИИ-нагрузок

ПМ.01 Разработка кода для обучения искусственного интеллекта

программы подготовки специалистов среднего звена
специальность
09.02.13 Интеграция решений с применением технологий искусственного интеллекта

Форма обучения - очная

Владивосток 2026

Рабочая программа МДК.01.05 Большие данные-платформы для поддержки ИИ-нагрузок разработана на основе федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.13 Интеграция решений с применением технологий искусственного интеллекта, квалификация специалист по работе с искусственным интеллектом, утвержденного приказом Министерства просвещения Российской Федерации от 24 декабря 2024 г. № 1025

Разработчик(и): С.Л. Шевырев, преподаватель

Рассмотрено и одобрено на заседании цикловой методической комиссии
Протокол № 9 от «20 » мая 2026 г.

Председатель ПЦК _____ П.В. Калашников

СОДЕРЖАНИЕ

1 **ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**

2 **СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**

3 **УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ**

4 **КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ**

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ МЕЖДИСЦИПЛИНАРНОГО КУРСА

1.1. Цель и место междисциплинарного курса в структуре образовательной программы

Междисциплинарный курс МДК.01.05 Большие данные-платформы для поддержки ИИ-нагрузок входит в профессиональный цикл учебного плана образовательной программы в структуру профессионального модуля ПМ.01 «Разработка кода для обучения искусственного интеллекта».

Цель: формирование у обучающихся профессиональных компетенций, необходимых для проектирования, разработки, тестирования и документирования программных модулей, предназначенных для реализации алгоритмов машинного обучения и других компонентов систем искусственного интеллекта.

1.2. Планируемые результаты освоения междисциплинарного курса

По итогам освоения курса, обучающиеся должны продемонстрировать результаты обучения, соотнесённые с результатами освоения ООП СПО, приведенные в таблице.

Код	Наименование видов деятельности и профессиональных компетенций
ВД 1	Разработка кода для обучения искусственного интеллекта.
ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
ПК 1.3	Оформлять программный код в соответствии с техническим заданием
ПК 1.4	Большие данные-платформы для поддержки ИИ-нагрузок
ПК 1.5	Выполнять отладку программных модулей с использованием специализированных программных средств
ПК 1.6	Большие данные-платформы для поддержки ИИ-нагрузок
ПК 1.7.	Составлять тестовые сценарии

В результате освоения междисциплинарного курса обучающийся должен:

Код ПК	Уметь	Знать	Владеть навыками
ПК 1.1	Анализировать требования технического задания для проектирования распределенных алгоритмов обработки потоковых и пакетных данных. Анализировать топологию кластера для минимизации сетевых задержек при передаче данных между узлами ИИ-систем.	Архитектурные шаблоны проектирования распределенных ИИ-приложений, а также математические и логические основы параллельного расчета градиентов и агрегации признаков.	Разработать логические схемы параллельной обработки данных. Оптимизировать вычислительную сложность распределенных алгоритмов на этапе проектирования. Разработать спецификации для пайплайнов подготовки обучающих выборок.
ПК 1.3	Анализировать написанный программный код на соответствие синтаксическим стандартам языков разработки Big Data (например, Python, Scala). Анализировать	Правила и стандарты форматирования распределенного кода, принципы чистого программирования в парадигме MapReduce, а также требования к документированию конфигураций Spark и	Разработать программный код распределенных модулей ИИ с соблюдением стандартов оформления. Оптимизировать структуру комментариев и конфигурационных параметров кода. Разработать регламенты

	структуру конфигурационных файлов для обеспечения воспроизводимости кода в различных средах.	Hadoop.	стилистического оформления кода для совместной разработки в команде.
ПК 1.4	Анализировать производительность экосистем Hadoop и Spark при выполнении ресурсоемких операций обучения ИИ-моделей. Анализировать паттерны доступа к распределенным файловым системам (HDFS) для оптимизации хранения сырых данных.	Архитектура платформ больших данных, методы горизонтального масштабирования вычислительных кластеров и принципы работы распределенных систем хранения данных.	Разработать конфигурацию Spark-сессии под конкретный объем ИИ-нагрузки. Оптимизировать параметры кэширования и распределения памяти в кластере. Разработать структуру хранения данных для ускорения дисковых операций ввода-вывода.
ПК 1.5	Анализировать логи и дампы памяти распределенных приложений (Spark UI, YARN logs) для поиска узких мест и падений задач. Анализировать метрики утилизации ресурсов кластера во время выполнения ИИ-нагрузок.	Методы удаленной отладки распределенного кода, функционал встроенных профилировщиков памяти и принципы поиска ошибок переполнения пямяти.	Разработать отладочные сценарии для изоляции ошибок на отдельных узлах кластера. Оптимизировать потребление памяти распределенными функциями в процессе отладки. Разработать систему логирования критических этапов обучения моделей.
ПК 1.6	Анализировать потоки данных в реальном времени (например, Kafka) для бесшовной интеграции со стриминговыми ИИ-моделями. Анализировать утилизацию графических процессоров (GPU) в рамках кластерной инфраструктуры платформ.	Технологии очередей сообщений, принципы работы NoSQL баз данных как хранилищ векторов (векторных эмбеддингов) и методы оркестрации тяжелых ИИ-задач.	Разработать архитектуру шины данных для потоковой трансляции признаков ИИ. Оптимизировать пропускную способность конвейера распределенной доставки данных. Разработать отказоустойчивые конфигурации для сервисов инференса моделей.
ПК 1.7	Анализировать граничные условия и аномалии в терабайтных массивах данных для формирования тестовых покрытий. Анализировать распределенные модули на устойчивость к отказам отдельных серверов.	Методологии модульного, интеграционного и нагрузочного тестирования распределенных ИИ-систем, а также генераторы синтетических данных.	Разработать автоматизированные тестовые сценарии для проверки корректности агрегации данных. Оптимизировать время выполнения тестов на больших объемах данных. Разработать наборы синтетических данных для проверки устойчивости ИИ-моделей к шуму.

2. СТРУКТУРА И СОДЕРЖАНИЕ МЕЖДИСЦИПЛИНАРНОГО КУРСА

2.1 Объем МДК и виды учебной работы

Вид учебной работы	Объем в часах	В т.ч. в форме практ. подготовки
Учебные занятия	84	42
Самостоятельная работа	34	
Консультации	2	
Промежуточная аттестация	4	
Всего	**124**	**42**

2.2. Тематический план и содержание МДК

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Коды компетенций, формированию которых способствует элемент программы
Раздел 1. Архитектура и инфраструктура платформ больших данных для искусственного интеллекта			
Тема 1.1. Распределенные системы хранения и экосистема Apache Hadoop	**Содержание**		
	Понятие больших данных (Big Data) и их специфика в задачах искусственного интеллекта (ИИ). Архитектура распределенной файловой системы Hadoop (HDFS — Hadoop Distributed File System) и планировщика ресурсов YARN (Yet Another Resource Negotiator). Принципы горизонтального масштабирования и обеспечения отказоустойчивости кластеров.	6	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	В том числе практических занятий	**6**	
	Развертывание и конфигурирование базовых компонентов HDFS. Загрузка, извлечение и управление распределенными файлами и наборами данных для ИИ-нагрузок через интерфейс командной строки.	6	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	Самостоятельная работа	5	
	Анализ паттернов доступа к данным и подготовка схемы оптимального размещения сырых датасетов на узлах кластера.	5	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
Тема 1.2. Парадигма MapReduce и концепция распределенных вычислений	**Содержание**	11	
	Принципы параллельной обработки информации MapReduce (отображение / свертка). Логика распределенного проектирования алгоритмов, параллельного расчета градиентов и агрегации признаков. Ограничения дисковых вычислений при обучении тяжелых ИИ-моделей.	6	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7

	В том числе практических занятий	6	
	Проектирование и декомпозиция алгоритма обработки текстовых данных в парадигме MapReduce. Написание и запуск базового скрипта очистки признаков.	6	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	Самостоятельная работа	5	
	Разработка логических схем параллельной предобработки структурированных и неструктурированных данных под требования технического задания.	5	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
Раздел 2. Высокопроизводительные распределенные вычисления в Apache Spark			
Тема 2. 1. Архитектура Apache Spark и обработка данных в оперативной памяти	**Содержание**	22	
	Архитектура Apache Spark, концепция вычислений в оперативной памяти (In-Memory). Устойчивые распределенные датасеты (RDD — Resilient Distributed Datasets) и оптимизированные структуры DataFrames. Управление памятью, ленивые вычисления (lazy evaluations) и направленные ациклические графы (DAG — Directed Acyclic Graph).	8	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	В том числе практических занятий	8	
	Настройка конфигурации Spark-сессии под ИИ-нагрузки. Программирование операций трансформации и действия над объектами DataFrames. Применение кэширования для ускорения повторного доступа к данным.	8	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	Самостоятельная работа	6	
	Оптимизация параметров распределения памяти в кластере и минимизация сетевых задержек при перемешивании данных (shuffling).	6	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
Тема 2.2. Машинное обучение на больших данных с помощью Spark MLlib	**Содержание**		
	Библиотека машинного обучения Spark MLlib: архитектура, встроенные распределенные алгоритмы классификации, регрессии и кластеризации. Создание масштабируемых конвейеров (Pipelines) машинного обучения. Подготовка, масштабирование и векторизация признаков на больших объемах данных.	8	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	В том числе практических занятий	8	
	Разработка распределенного пайплайна обучения модели ИИ на многогигабайтном датасете. Оценка качества и валидация моделей на распределенном кластере.	8	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	Самостоятельная работа	6	
	Экспериментальный анализ влияния изменения гиперпараметров	6	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5;

	алгоритмов Spark MLlib на общую скорость работы моделей.		ПК 1.6; ПК 1.7
Раздел 3. Потоковая обработка, хранение эмбеддингов и оркестрация ИИ-нагрузок			
Тема 3. 1. Архитектура очередей сообщений и потоковая обработка данных в реальном времени	**Содержание**	12	
	Платформа распределенного обмена сообщениями Apache Kafka: топики, продюсеры, консьюмеры и партиционирование. Концепция потоковой обработки (Streaming) и интеграция брокеров сообщений со стриминговыми ИИ-моделями для инференса «на лету».	4	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	В том числе практических занятий	4	
	Разработка шины данных на базе Apache Kafka для потоковой трансляции признаков ИИ. Подключение Spark Streaming для обработки входящего потока.	4	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	Самостоятельная работа	4	
	Оптимизация пропускной способности конвейера распределенной доставки данных и проектирование отказоустойчивой конфигурации.	4	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
Тема 3.2. Специализированные NoSQL базы данных и векторные хранилища	**Содержание**		
	Классификация нереляционных баз данных (NoSQL — Not Only SQL). Нереляционные СУБД (Системы управления базами данных) как хранилища векторов и высокоразмерных векторных эмбеддингов (Vector Databases). Индексация больших объемов признаков для быстрого семантического поиска.	4	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	В том числе практических занятий	4	
	Проектирование структуры хранения векторов в специализированной СУБД. Выполнение операций быстрого поиска ближайших соседей по базе эмбеддингов.	4	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	Самостоятельная работа	4	
	Сравнительный анализ производительности различных типов индексации векторных данных под требования высокой интенсивности запросов.	4	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
Тема 3.3. Отладка, тестирование и оркестрация распределенных ИИ-систем	**Содержание**		
	Методологии модульного, интеграционного и нагрузочного тестирования распределенных приложений. Инструменты оркестрации ИИ-задач (например, Apache Airflow). Инструменты мониторинга и отладки (Spark UI, YARN logs), методы локализации	6	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7

	ошибок нехватки памяти (OOM — Out of Memory) и дисбаланса данных (data skew).		
	В том числе практических занятий	6	
	Анализ логов распределенных приложений и профилирование памяти для поиска узких мест. Составление автоматизированных тестовых сценариев и генерация синтетических данных для нагрузочного тестирования распределенных ИИ-модулей.	6	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7
	Самостоятельная работа	4	
	Разработка отладочных сценариев для изоляции ошибок на отдельных узлах и проектирование комплексной системы логирования критических этапов обучения моделей.	4	ПК 1.1; ПК 1.3; ПК 1.4; ПК 1.5; ПК 1.6; ПК 1.7

3. УСЛОВИЯ РЕАЛИЗАЦИИ МЕЖДИСЦИПЛИНАРНОГО КУРСА

3.1. Материально-техническое обеспечение

Для реализации программы МДК.01.05 «Большие данные-платформы для поддержки ИИ-нагрузок» предусмотрено наличие следующих помещений:

Лаборатория организации и принципов построения информационных систем.

Основное оборудование: Компьютеризированные рабочие места обучающихся с выходом в сеть "Интернет" - 20 шт., стулья - 20 шт., комплект мебели (стол и стул) для преподавателя - 1 шт., графические станции ПК №2 WorkStation WS 917 20 шт., автоматизированное рабочее место преподавателя ПК №2 WorkStation WS 917, графический планшет Wacom - 20 шт., мультимедийное оборудование - 1 шт., телевизор, экран, многофункциональное устройство (МФУ) формат А3 -1 шт., доска маркерная - 1 шт.

Программное обеспечение: PyCharm, Visual Studio Code, IntelliJ IDEA, JetBrains Rider, Python (Anaconda3) с TensorFlow/PyTorch, Microsoft Visio, Draw.io, Figma, Figma, Adobe Illustrator, MySQL Workbench, HeidiSQL, MySQL, PostgreSQL, Git, Docker Desktop, Microsoft Project, Draw.io, Microsoft SQL Server Management Studio, Microsoft Office.

Помещение для самостоятельной и воспитательной работы обучающихся, оснащенное компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации.

Основное оборудование: Рабочие места на базе компьютерной техники с возможностью подключения к информационно-телекоммуникационной сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ВВГУ: комплекты учебной мебели (столы и стулья) – 19 шт., персональные компьютеры (облачные мониторы) - 19 шт; доска маркерная - 1шт., лазерный копир-принтер-сканер с 2 лотками Xerox WorkCentre 3345 DNI.

Программное обеспечение: Microsoft Windows 7 Professional; Microsoft Office ProPlus 2010 Russian Acdmc; СПС КонсультантЮрист: Версия Проф; Adobe Acrobat Reader; Google Chrome; Adobe Flash Player; 7-Zip 18.01 (x64).

3.2. Учебно-методическое обеспечение

Для реализации программы учебной дисциплины библиотечный фонд ВВГУ укомплектован печатными и электронными изданиями.

Обучающиеся из числа инвалидов и лиц с ограниченными возможностями здоровья обеспечены печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

Основные источники

1. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 273 с. — (Профессиональное образование). — ISBN 978-5-534-20362-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/562355

2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 404 с. — (Профессиональное образование).
— ISBN 978-5-534-19506-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/566739

3. Соколова, В. В. Разработка мобильных приложений : учебное пособие для среднего профессионального образования / В. В. Соколова. — Москва : Издательство Юрайт, 2024. —160 с. — (Профессиональное образование). — ISBN 978-5-534-16868-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/542342

Дополнительные источники

1. Черпаков, И. В. Основы программирования : учебник и практикум для среднего профессионального образования / И. В. Черпаков. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 196 с. — (Профессиональное образование). — ISBN 978-5-534- 18760-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/561922

2. Зыков, С. В. Объектно-ориентированное программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 151 с. — (Высшее образование). — ISBN 978-5-534-16941-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/537385

3. Станкевич, Л. А. Интеллектуальные системы и технологии : учебник и практикум для среднего профессионального образования / Л. А. Станкевич. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 478 с. — (Профессиональное образование). — ISBN 978-5-534-20364-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/566524

Электронные ресурсы

1. Информационно-справочная система «Консультант Плюс» – http://www.consultant.ru/

2. Профессиональная база данных: "Открытая база ГОСТов"/ Режим доступа: http://standartgost.ru/, доступ свободный

3. Электронная библиотечная система «РУКОНТ» - Режим доступа: https://lib.rucont.ru/

4. Электронная библиотечная система ZNANIUM.COM - Режим доступа: https://znanium.com/

5. Электронно-библиотечная система издательства "Юрайт" - Режим доступа: https://urait.ru/

6. Электронно-библиотечная система издательства "Лань" - Режим доступа: https://e.lanbook.com/

7. Президентская библиотека им. Б.Н. Ельцина (база данных различных профессиональных областей) Режим доступа: https://www.prlib.ru/, доступ свободный

4 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ИЗУЧЕНИЮ И РЕАЛИЗАЦИИ МЕЖДИСЦИПЛИНАРНОГО КУРСА

Начиная изучение дисциплины, студенту необходимо:

- ознакомиться с программой, изучить список рекомендуемой литературы;
- внимательно разобраться в структуре курса, в системе распределения учебного материала по видам занятий, формам контроля, чтобы иметь представление о курсе в целом.

Самостоятельная работа студентов (СРС) — это деятельность учащихся, которую они совершают без непосредственной помощи и указаний преподавателя, руководствуясь сформировавшимися ранее представлениями о порядке и правильности выполнения операций. Цель СРС в процессе обучения заключается, как в усвоении знаний, так и в формировании умений и навыков по их использованию в новых условиях на новом учебном материале. Самостоятельная работа призвана обеспечивать возможность осуществления студентами самостоятельной познавательной деятельности в обучении, и является видом учебного труда, способствующего формированию у студентов самостоятельности. Кроме того, для расширения и углубления знаний по дисциплине целесообразно использовать: публикации в тематических журналах; полнотекстовые базы данных библиотеки; имеющиеся в библиотеках вуза. Успешное освоение дисциплины предполагает активную работу студентов на всех занятиях аудиторной формы: лекций и практических занятий, выполнение аттестационный мероприятий, эффективную самостоятельную работу.

4.1 Методические рекомендации обучающимся по обеспечению самостоятельной работы

Самостоятельная работа проводится с целью: систематизации, закрепления, углубления и расширения теоретических знаний и практических умений, приобретаемых студентами в ходе аудиторных занятий; формирования умений использовать специальную литературу; развития познавательных способностей и активности обучающихся; формирования самостоятельности мышления, способности к саморазвитию, самосовершенствованию и самореализации; развития исследовательских умений.

Для лучшего освоения материала и систематизации знаний по дисциплине необходимо отвечать на вопросы для самоконтроля. Во время самостоятельной проработки лекционного материала особое внимание следует уделять возникшим вопросам, непонятным терминам, спорным точкам зрения. В случае необходимости обращаться к преподавателю за консультацией. Студент должен четко уяснить, что именно с лекции начинается его подготовка к практическому занятию. Вместе с тем, лекция лишь организует мыслительную деятельность, но не обеспечивает глубину усвоения программного материала.

При подготовке к практическому занятию особое внимание необходимо обратить на содержание основных положений и выводов, объяснение явлений и фактов, уяснение практического приложения рассматриваемых теоретических вопросов. В процессе этой работы студент должен стремиться понять и запомнить основные положения рассматриваемого материала, примеры, поясняющие его, а также разобраться в иллюстративном материале. В процессе подготовки рекомендуется взаимное обсуждение материала, во время которого закрепляются знания, а также приобретается практика в изложении и разъяснении полученных знаний, развивается речь.

При необходимости следует обращаться за консультацией к преподавателю. Перед консультацией, необходимо хорошо продумать вопросы, которые требуют разъяснения.

В процессе изучения дисциплины студенту необходимо ориентироваться на самостоятельную проработку лекционного материала, подготовку к практическим занятиям, выполнение тестов, кейсовых заданий, самостоятельное изучение некоторых разделов курса. Для проведения занятий лекционного типа используются

учебно-наглядные пособия в форме презентационных материалов, обеспечивающих тематические иллюстрации, соответствующие темам лекций.

4.2 Особенности организации обучения для лиц с ограниченными возможностями здоровья и инвалидов

При необходимости обучающимся из числа лиц с ограниченными возможностями здоровья и инвалидов (по заявлению обучающегося) предоставляется учебная информация в доступных формах с учетом их индивидуальных психофизических особенностей:

для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания, консультации и др.

для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания, консультации и др.

для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; индивидуальные задания, консультации и др.

5 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ КУРСА

Код ПК	Критерии оценки результата (показатели освоенности компетенций)	Формы контроля и методы оценки
ПК.1.1	«Отлично»: Самостоятельно строит оптимальные алгоритмы параллельной обработки данных (MapReduce, Spark DAG). Учитывает специфику ИИ-нагрузок. Оптимизирует использование памяти и процессора. «Хорошо»: Правильно строит алгоритмы обработки данных. Допускает незначительные логические ошибки, не влияющие на общий результат. Требуются небольшие подсказки при оптимизации под ИИ.«Удовлетворительно»: Строит базовые линейные алгоритмы. Затрудняется в реализации распределенной логики. Алгоритм работает неэффективно на больших объемах данных. «Отлично»: Самостоятельно строит оптимальные алгоритмы параллельной обработки данных (MapReduce, Spark DAG). Учитывает специфику ИИ-нагрузок. Оптимизирует использование памяти и процессора.«Хорошо»: Правильно строит алгоритмы обработки данных. Допускает незначительные логические ошибки, не влияющие на общий результат. Требуются небольшие подсказки при оптимизации под ИИ. «Удовлетворительно»: Строит базовые линейные алгоритмы. Затрудняется в реализации распределенной логики. Алгоритм работает неэффективно на больших объемах данных.	Защита отчета по практической/лабораторной работе, устное собеседование.
ПК 1.3	«Отлично»: Код полностью соответствует ТЗ и стандартам (PEP 8, Google Style Guide). Структура проекта идеальна. Присутствуют исчерпывающие комментарии и документация к API. «Хорошо»: Код выполняет требования ТЗ. Стандарты оформления соблюдены. Есть единичные нарушения в именовании переменных или структуре логических блоков. «Удовлетворительно»: Код решает задачу лишь	Защита отчета (проверка листинга кода), устное собеседование.

	частично. Оформление хаотично. Отсутствуют комментарии. Нарушены требования ТЗ к форматам ввода-вывода данных.	
ПК 1.4	«Отлично»: Свободно обосновывает выбор архитектуры платформы под конкретную ИИ-задачу. Безошибочно конфигурирует хранилища и вычислительные кластеры. «Хорошо»: Понимает принципы работы платформ больших данных. Успешно настраивает базовую конфигурацию среды для ИИ. Допускает мелкие ошибки в теории архитектур. «Удовлетворительно»: Знает основные компоненты экосистемы (HDFS, Spark). Способен развернуть преднастроенный шаблон платформы по инструкции, но не может его изменить.	Прохождение тестов, устное собеседование, защита отчета.:
ПК 1.5	«Отлично»: Быстро находит скрытые ошибки (утечки памяти, перекос данных/Data Skew в Spark) с помощью UI-инструментов и логов. Проводит профилирование ИИ-нагрузок. «Хорошо»: Уверенно локализует и исправляет синтаксические и явные логические ошибки в распределенном коде. Пользуется стандартными логами отладки. «Удовлетворительно»: Находит ошибки только с помощью преподавателя или базового вывода print/console.log. Исправляет дефекты методом подбора, не понимая их сути.	Защита отчета (демонстрация логов и процесса отладки), устное собеседование.
ПК 1.6	«Отлично»: Успешно интегрирует пайплайны данных с моделями ИИ (например, Spark MLlib, TensorFlow на кластере). Настраивает динамическое масштабирование ресурсов. «Хорошо»: Настраивает передачу данных из Big Data хранилища в ИИ-модуль. Ошибается в тонкой настройке параметров масштабирования кластера. «Удовлетворительно»: Может запустить готовый ИИ-скрипт на локальной копии данных. Не умеет масштабировать нагрузку на реальный кластер больших данных.	Защита отчета, устное собеседование.
ПК 1.7	«Отлично»: Разрабатывает комплексные сценарии тестирования (модульные, нагрузочные, интеграционные). Учитывает граничные условия и специфику терабайтных объемов данных. «Хорошо»: Составляет корректные функциональные тесты для проверки программных модулей. Покрывает основные ветки алгоритма, но упускает нагрузочные тесты. «Удовлетворительно»: Пишет простейшие позитивные тесты (Happy Path). Не проверяет реакцию системы на некорректные данные или падение узлов кластера.	Защита отчета (документ тест-плана), прохождение тестов.

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Колледж информационных и креативных технологий

КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА

для проведения текущего контроля и промежуточной аттестации

по **МДК.01.05 Большие данные-платформы для поддержки ИИ-нагрузок**

ПМ.01 Разработка кода для обучения искусственного интеллекта

программы подготовки специалистов среднего звена
09.02.13 Интеграция решений с применением технологий искусственного интеллекта

Форма обучения: очная

Владивосток 2026

1. Общие сведения

Контрольно-оценочные средства (далее – КОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу МДК.01.05 «Техническая документация в ИИ-проектах и анализа данных» в соответствии с техническим заданием и оценка их качества

КОС включают в себя контрольные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине, которая проводится в форме дифференцированного зачёта (с использованием оценочного средства - устный опрос в форме ответов на вопросы билетов, устный опрос в форме собеседования, выполнение письменных заданий, тестирование и т.д.).

2. Планируемые результаты обучения, обеспечивающие результаты освоения образовательной программы

Код ПК	Код результата обучения	Наименование результата обучения
ПК 1.1	З1	Основные методы параллельной обработки данных и принципы построения направленных ациклических графов (DAG).
	З2	Специфика алгоритмов стриминговой (потоковой) и пакетной обработки больших данных для обучения ИИ-моделей.
	У1	Проектировать логическую структуру распределенных вычислительных модулей на основе требований технического задания.
	У2	Анализировать сложность алгоритмов и оценивать их масштабируемость при возрастании объема обрабатываемой информации.
	П1	Навыки декомпозиции комплексных задач ИИ на отдельные этапы конвейера обработки данных (Data Pipeline).
	П2	Опыт построения блок-схем и алгоритмических моделей распределенной фильтрации, агрегации и трансформации признаков.
ПК 1.3	З1	Правила оформления кода и стандарты PEP 8 (для Python) или Google Style Guide (для Scala/Java) в проектах Big Data.
	З2	Принципы документирования программного кода (Docstrings) и форматы описания API для интеграции с ИИ-сервисами.
	У1	Оформлять исходный код программных модулей с соблюдением требований ТЗ к структуре каталогов и чистоте синтаксиса.
	У2	Комментировать логику работы распределенных функций и методов подготовки данных.
	П1	Инструменты автоматической проверки качества и форматирования кода (Linter, Black, Flake8).
	П2	Технологии контроля версий программного кода (Git) при совместной разработке ИИ-платформ.
ПК 1.4	З1	Архитектура и компоненты современных платформ хранения больших данных (HDFS, S3-совместимые объектные хранилища).
	З2	Различия между реляционными СУБД и NoSQL-решениями, применяемыми для хранения сырых и структурированных данных ИИ.
	У1	Выбирать оптимальную конфигурацию платформы больших данных под конкретный тип ИИ-нагрузки (компьютерное зрение, NLP).
	У2	Развертывать локальные и распределенные вычислительные среды (Apache Spark, Hadoop) по заданным спецификациям.
	П1	Навыки конфигурирования кластерных ресурсов и параметров

		оперативной памяти для обработки терабайтных массивов.
	П2	Методы организации безопасного доступа к распределенным хранилищам данных.
ПК 1.5	З1	Специализированные программные средства и UI-инструменты мониторинга распределенных вычислений (Spark UI, Yarn ResourceManager).
	З2	Типовые ошибки распределенного выполнения кода (утечки памяти, перекос данных / Data Skew, сетевые задержки).
	У1	Локализовывать программные сбои по системным логам кластера и журналам ошибок.
	У2	Анализировать профиль выполнения задач для устранения узких мест в коде подготовки данных.
	П1	Инструменты логирования работы программных модулей и отслеживания метрик производительности.
	П2	Техники интерактивной пошаговой отладки кода ИИ-нагрузок в изолированных контейнерах или блокнотах (Jupyter).
ПК 1.6	З1	Архитектурные шаблоны интеграции платформ больших данных со фреймворками машинного обучения (PyTorch, TensorFlow).
	З2	Принципы построения ETL/ELT процессов интеграции данных из разрозненных источников в единое озеро данных (Data Lake).
	У1	Настраивать потоковую передачу данных из очередей сообщений (Kafka) в модули инференса ИИ-моделей.
	У2	Оптимизировать форматы хранения данных (Parquet, ORC, Avro) для ускорения процедур обучения нейросетей.
	П1	Навыки развертывания конвейеров непрерывной интеграции данных (MLOps базового уровня).
	П2	Методы масштабирования вычислительных узлов платформы при изменении интенсивности входящего потока данных.
ПК 1.7	З1	Виды тестирования систем больших данных (функциональное, интеграционное, нагрузочное, стресс-тестирование).
	З2	Методология создания синтетических наборов данных (Mock-данных) для верификации ИИ-компонентов.
	У1	Формировать тестовые сценарии с проверкой граничных условий и валидацией схем данных на входе.
	У2	Проектировать тесты для оценки отказоустойчивости платформы при имитации падения отдельных узлов кластера.
	П1	Инструменты автоматизации модульного тестирования (PyTest, ScalaTest).
	П2	Технологии фиксации результатов тестирования и формирования отчетов о покрытии кода тестами.

3. Описание процедуры оценивания

Результаты обучения по модулю, уровень сформированности компетенций оцениваются по четырёх бальной шкале оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» (по бальной системе. Максимальная сумма баллов по дисциплине равна 100 баллов.)

Текущая аттестация по модулю проводится с целью систематической проверки достижений обучающихся. Объектами оценивания являются: степень усвоения теоретических знаний, уровень овладения практическими умениями и навыками по всем видам учебной работы, качество выполнения самостоятельной работы, учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине).

При проведении промежуточной аттестации оценивается достижение студентом запланированных по модулю результатов обучения, обеспечивающих результаты

освоения образовательной программы в целом. Оценка на зачете / экзамене выставляется с учетом оценок, полученных при прохождении текущей аттестации.

Критерии оценивания устного ответа

(оценочные средства: контрольная работа, собеседование, устное сообщение.)

5 баллов - ответ показывает прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа; умение приводить примеры современных проблем изучаемой области.

4 балла - ответ, обнаруживающий прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа. Однако допускается одна - две неточности в ответе.

3 балла – ответ, свидетельствующий в основном о знании процессов изучаемой предметной области, отличающийся недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры; недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа; неумение привести пример развития ситуации, провести связь с другими аспектами изучаемой области.

2 балла – ответ, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные ошибки в содержании ответа; незнание современной проблематики изучаемой области.

Критерии оценивания письменной работы

(оценочные средства: реферат, конспект, контрольная работа, письменный отчет по практической работе, портфолио, доклад (сообщение), в том числе выполненный в форме презентации, творческое задание, курсовая работа).

5 баллов - студент выразил своё мнение по сформулированной проблеме, аргументировал его, точно определив ее содержание и составляющие. Проблема раскрыта полностью, выводы обоснованы. Приведены данные отечественной и зарубежной литературы, статистические сведения, информация нормативно-правового характера. Студент владеет навыком самостоятельной работы по заданной теме; методами и приемами анализа теоретических и/или практических аспектов изучаемой области. Фактических ошибок, связанных с пониманием проблемы, нет; графически работа оформлена правильно.

4 балла - работа характеризуется смысловой цельностью, связностью и последовательностью изложения; допущено не более 1 ошибки при объяснении смысла или содержания проблемы. Проблема раскрыта. Не все выводы сделаны и/или обоснованы. Для аргументации приводятся данные отечественных и зарубежных авторов. Продемонстрированы исследовательские умения и навыки. Фактических ошибок, связанных с пониманием проблемы, нет. Допущены одна-две ошибки в оформлении работы.

3 балла – студент проводит достаточно самостоятельный анализ основных этапов и смысловых составляющих проблемы; понимает базовые основы и теоретическое обоснование выбранной темы. Проблема раскрыта не полностью. Выводы не сделаны и/или выводы не обоснованы Проведен анализ проблемы без привлечения дополнительной литературы. Допущено не более 2 ошибок в смысле или содержании проблемы, оформлении работы.

2 балла - работа представляет собой пересказанный или полностью переписанный исходный текст без каких бы то ни было комментариев, анализа. Не раскрыта структура и теоретическая составляющая темы. Проблема не раскрыта. Выводы отсутствуют. Допущено три или более трех ошибок в смысловом содержании раскрываемой проблемы, в оформлении работы.

Критерии оценивания тестового задания

Оценка	Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Количество правильных ответов	91 % и ≥	от 81% до 90,9 %	не менее 70%	менее 70%

Критерии выставления оценки студенту на зачете/ экзамене

(оценочные средства: устный опрос в форме ответов на вопросы билетов, устный опрос в форме собеседования, выполнение письменных разноуровневых задач и заданий, комплексная расчетно-графическая работа, творческое задание, кейс-задача, портфолио, проект и т.п.)

Оценка по промежуточной аттестации	Характеристика качества сформированности компетенций
«зачтено» / «отлично»	Студент демонстрирует сформированность дисциплинарных компетенций на продвинутом уровне: обнаруживает всестороннее, систематическое и глубокое знание учебного материала, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
«зачтено» / «хорошо»	Студент демонстрирует сформированность дисциплинарных компетенций на базовом уровне: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
«зачтено» / «удовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на пороговом уровне: имеет знания только основного материала, но не усвоил его деталей, в ходе контрольных мероприятий допускаются значительные ошибки, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ, при оперировании знаниями и умениями при их переносе на новые ситуации.
«не зачтено» / «неудовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на уровне ниже порогового: выявляется полное или практически полное отсутствие знаний значительной части программного материала, студент допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы, умения и навыки не сформированы.

4. Примеры оценочных средств для проведения промежуточной аттестации

Вопросы для дифференцированного зачета

1. Государственные стандарты (ГОСТ) серии 19 и 34: назначение, область применения и ключевые отличия при проектировании ИИ-систем.
2. Международная методология CRISP-DM: цели, основные этапы и виды формируемой документации на каждом шаге.
3. Структура и обязательные разделы технического задания (ТЗ) на разработку программного модуля анализа данных.
4. Специфика формулирования требований к бизнес-логике и ограничениям предметной области в ИИ-проектах.
5. Правила документирования требований к качеству, объему, формату и полноте входящих наборов данных.
6. Назначение и структура документа Спецификация набора данных (Dataset Card): ключевые разделы и правила заполнения.
7. Метрики качества моделей машинного обучения (Accuracy, Precision, Recall, F1-score): правила их фиксации и обоснования в ТЗ.
8. Документирование требований к вычислительной инфраструктуре, аппаратной части (CPU/GPU) и времени отклика ИИ-моделей.
9. Способы декомпозиции комплексных задач интеллектуального анализа данных на отдельные функциональные блоки.
10. Графические нотации для описания алгоритмов: правила построения блок-схем в соответствии с ГОСТ 19.701-90.
11. Применение Унифицированного языка моделирования (UML) в ИИ-проектах: типы диаграмм, используемые для описания архитектуры.
12. Документирование диаграмм компонентов (Component Diagram) для интеграции ИИ-модуля в существующее программное обеспечение.
13. Правила технического описания пайплайнов процессов извлечения, преобразования и загрузки данных (ETL).
14. Документирование этапа конструирования признаков (Feature Engineering): фиксация математического описания и логики преобразований.
15. Архитектурные шаблоны построения алгоритмов классификации и регрессии: способы их отражения в проектной документации.
16. Особенности документирования архитектуры глубоких нейронных сетей: описание слоев, функций активации и гиперпараметров.
17. Методы теоретической оценки и документирования вычислительной сложности алгоритмов (О-нотация) на этапе проектирования.
18. Порядок согласования, утверждения и внесения изменений в ТЗ на разработку систем искусственного интеллекта.
19. Документирование рисков проекта, связанных со смещением (bias) данных, недообучением и переобучением моделей.
20. Анализ исходного ТЗ на полноту и непротиворечивость требований: составление экспертного чек-листа проверки.
21. Стандарты чистой разработки программного обеспечения: назначение руководства по стилю кода PEP 8 для языка Python.
22. Правила и синтаксис ведения строк документации (Docstrings) для функций, классов и модулей.
23. Стили оформления Docstrings (Google Style, NumPy Style, Sphinx Style) и их основные различия.
24. Особенности документирования и правила оформления интерактивных

блокнотов (Jupyter-ноутбуков) для передачи в продакшн.

25. Понятие технического долга в коде ИИ-систем и способы его документирования в процессе рефакторинга.

26. Инструменты автоматической генерации технической документации из исходного кода (на примере утилиты Sphinx).

27. Правила ведения файлов конфигурации зависимостей (requirements.txt, environment.yml) и фиксации версий библиотек.

28. Документирование интерфейсов программирования приложений (API): назначение, принципы и стандарты REST-архитектуры.

29. Спецификация OpenAPI и инструмент Swagger: назначение, структура YAML/JSON-файлов описания API.

30. Правила документирования форматов обмена данными (JSON/YAML) между ИИ-модулем и внешними системами.

31. Описание кодов ответов сервера и структуры ошибок ИИ-сервиса в технической документации к API.

32. Применение систем контроля версий (Git) для совместной работы над кодом и сопутствующей документацией ИИ-проекта.

33. Правила документирования коммитов (Conventional Commits) и ведения журнала изменений (Changelog).

34. Линтеры (Flake8, Pylint) и автоформаттеры (Black): роль в автоматизированном контроле качества кода и документации.

35. Проведение аудита кода (code review): правила документирования замечаний и рекомендаций к программным модулям ИИ.

36. Назначение, структура и правила составления Программы и методики испытаний (ПМИ) для ИИ-систем.

37. Документирование процесса тестирования: фиксация плановых и фактических метрик точности моделей.

38. Документирование программных ограничений, граничных условий и аномалий, выявленных в ходе тестирования ИИ-модуля.

39. Понятие контейнеризации: структура и правила документирования конфигурационных файлов Docker (Dockerfile, docker-compose.yml).

40. Документирование пайплайнов непрерывной интеграции и доставки (CI/CD) для автоматического тестирования и деплоя ИИ-моделей.

41. Назначение и структура Спецификации модели (Model Card): документирование ограничений, этических аспектов и условий применения ИИ.

42. Структура и правила написания ознакомительного файла (README.md) для репозитория проекта ИИ.

43. Разработка Руководства системного администратора: правила описания шагов по развертыванию и администрированию ИИ-модуля.

44. Разработка Руководства пользователя: требования к описанию интерфейса, входных параметров и интерпретации результатов работы ИИ.

45. Правила фиксации и документирования инцидентов, багов и сбоев в работе ИИ-моделей на этапе промышленной эксплуатации.

Экзаменационные билеты

Билет 1

1. Государственные стандарты (ГОСТ) серии 19 и 34: назначение, область применения и ключевые отличия при проектировании ИИ-систем.
2. Инструменты автоматической генерации технической документации из исходного кода

(на примере утилиты Sphinx).

Билет 2

1. Международная методология CRISP-DM: цели, основные этапы и виды формируемой документации на каждом шаге.
2. Спецификация OpenAPI и инструмент Swagger: назначение, структура YAML/JSON-файлов описания API.

Билет 3

1. Назначение и структура документа Спецификация набора данных (Dataset Card): ключевые разделы и правила заполнения.
2. Стандарты чистой разработки программного обеспечения: назначение руководства по стилю кода PEP 8 для языка Python.

Билет 4

1. Метрики качества моделей машинного обучения (Accuracy, Precision, Recall, F1-score): правила их фиксации и обоснования в ТЗ.
2. Правила ведения файлов конфигурации зависимостей (requirements.txt, environment.yml) и фиксации версий библиотек.

Билет 5

1. Графические нотации для описания алгоритмов: правила построения блок-схем в соответствии с ГОСТ 19.701-90.
2. Назначение, структура и правила составления Программы и методики испытаний (ПМИ) для ИИ-систем.

Билет 6

1. Применение Унифицированного языка моделирования (UML) в ИИ-проектах: типы диаграмм, используемые для описания архитектуры.
2. Понятие контейнеризации: структура и правила документирования конфигурационных файлов Docker (Dockerfile, docker-compose.yml).

Билет 7

1. Правила технического описания пайплайнов процессов извлечения, преобразования и загрузки данных (ETL).
2. Назначение и структура Спецификации модели (Model Card): документирование ограничений, этических аспектов и условий применения ИИ.

Билет 8

1. Методы теоретической оценки и документирования вычислительной сложности алгоритмов (O-нотация) на этапе проектирования.
2. Структура и правила написания ознакомительного файла (README.md) для репозитория проекта ИИ.

Билет 9

1. Структура и обязательные разделы технического задания (ТЗ) на разработку программного модуля анализа данных.
2. Особенности документирования и правила оформления интерактивных блокнотов (Jupyter-ноутбуков) для передачи в продакшн.

Билет 10

1. Особенности документирования архитектуры глубоких нейронных сетей: описание слоев, функций активации и гиперпараметров.
2. Разработка Руководства системного администратора: правила описания шагов по развертыванию и администрированию ИИ-модуля.