Федеральное государственное автономное образовательное учреждение высшего образования
«Российский университет дружбы народов имени Патриса Лумумбы»

Факультет искусственного интеллекта
(наименование основного учебного подразделения (ОУП)-разработчика ОП ВО)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

HADOOP, SPARK
(наименование дисциплины/модуля)

Рекомендована МССН для направлений подготовки:

02.03.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ;
09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА
(код и наименование направления подготовки/специальности)

Освоение дисциплины ведется в рамках реализации основной профессиональной образовательной программы высшего образования (ОП ВО):

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: РАЗРАБОТКА И ОБУЧЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ
(наименование (профиль/специализация) ОП ВО)

2026 г.

1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Hadoop, SPARK» входит в программу бакалавриата «Искусственный интеллект: разработка и обучение интеллектуальных систем» по направлениям подготовки 02.03.02 Фундаментальная информатика и информационные технологии и 09.03.03 Прикладная информатика, и изучается в 5 семестре 3 курса. Дисциплину реализует Кафедра прикладного искусственного интеллекта. Дисциплина состоит из 4 разделов и 34 тем и направлена на изучение технологий распределённой обработки и хранения больших данных на основе экосистемы Apache Hadoop (HDFS, YARN, MapReduce, Hive) и Apache Spark (RDD, DataFrame, Spark SQL, Spark MLlib, Structured Streaming): архитектуры распределённых кластеров, принципов горизонтального масштабирования, паттернов пакетной и потоковой обработки данных, а также применения этих технологий для построения масштабируемых пайплайнов подготовки данных, обучения моделей машинного обучения на больших объёмах данных и проектирования распределённых хранилищ для ИИ-систем.

Целью освоения дисциплины является формирование у студентов системных знаний и практических навыков работы с технологиями распределённой обработки больших данных Hadoop и Spark, включая способность проектировать архитектуру распределённых хранилищ, реализовывать ETL-пайплайны и аналитические запросы на больших объёмах данных, применять Spark MLlib для распределённого обучения моделей, организовывать потоковую обработку данных, а также обосновывать выбор технологического стека для задач ИИ с учётом требований к масштабируемости, производительности и отказоустойчивости.

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «Hadoop, SPARK» направлено на формирование у обучающихся следующих компетенций (части компетенций):

Таблица 2.1. Перечень компетенций, формируемых у обучающихся при освоении дисциплины (результаты освоения дисциплины)

Шифр	**Компетенция**	**Индикаторы достижения компетенции** (в рамках данной дисциплины)
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.3 Владеет навыками научного поиска и практической работы с информационными источниками; методами принятия решений;
ОПК-2	Способен понимать принципы работы современных информационных технологий и применять компьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности	ОПК-2.2 Умеет применять современное программное обеспечение (в том числе отечественного происхождения), фреймворки машинного обучения и инструменты обработки данных для решения задач в области ИИ; ОПК-2.3 Владеет навыками использования вычислительных методов, включая массово-параллельные вычисления на GPU, для обучения и развёртывания моделей ИИ;
ОПК-7	Способен решать задачи профессиональной деятельности на основе информационной культуры, применяя методы сбора, обработки, анализа и интерпретации данных с	ОПК-7.2 Умеет осуществлять сбор данных из различных источников, проводить разведочный анализ данных (EDA), статистический анализ, визуализацию, работать с распределёнными системами хранения и обработки данных;

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
	использованием информационно-коммуникационных технологий	
ПК-2	Способен проектировать архитектуру информационных систем с компонентами ИИ, разрабатывать прототипы и базы данных таких систем	ПК-2.1 Проектирует архитектуру ИС с компонентами ИИ, выбирает архитектурные паттерны и технологический стек; ПК-2.3 Проектирует и разрабатывает БД ИС с элементами ИИ, обеспечивает управление доступом к данным;
BD-3	Способен организовывать хранение данных, выбирая адекватные технологические решения	BD-3.1 Разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ с применением различных технологий хранения структурированных данных, оценивает качество построенных прикладных решений; BD-3.2 Разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ с применением различных технологий хранения неструктурированных данных, оценивает качество;
BD-4	Способен применять различные модели и (или) технологии обработки больших данных	BD-4.1 Осуществляет выбор технологий обработки больших данных, приемлемых для создания прикладной системы ИИ с заданными требованиями; BD-4.2 Разрабатывает и отлаживает прикладные решения с элементами ИИ с применением различных технологий обработки данных; BD-4.3 Тестирует, испытывает и оценивает качество решений с элементами ИИ, реализованных с использованием технологий обработки данных;
PL-1	Способен применять язык программирования Python для решения задач в области ИИ	PL-1.3 Разрабатывает и поддерживает системы обработки больших данных различной степени сложности;

3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «Hadoop, SPARK» относится к обязательной части блока 1 «Дисциплины (модули)» образовательной программы высшего образования.

В рамках образовательной программы высшего образования обучающиеся также осваивают другие дисциплины и/или практики, способствующие достижению запланированных результатов освоения дисциплины «Hadoop, SPARK».

Таблица 3.1. Перечень компонентов ОП ВО, способствующих достижению запланированных результатов освоения дисциплины

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	Линейная алгебра; Дискретная математика; Математический анализ; Теория вероятностей и математическая статистика; Статистические методы и первичный анализ данных; Алгоритмы и структуры данных; Введение в базы данных;	Преддипломная практика;
ОПК-2	Способен понимать принципы работы современных информационных	История и теория программирования; Введение в искусственный интеллект;	Массово-параллельные вычисления в машинном обучении (GPU); Основы глубокого обучения;

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
	технологий и применять компьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности	Программирование на языке Python;	Методы машинного обучения; Нейронные сети;
ОПК-7	Способен решать задачи профессиональной деятельности на основе информационной культуры, применяя методы сбора, обработки, анализа и интерпретации данных с использованием информационно-коммуникационных технологий	Технологическая (проектно-технологическая) практика (учебная); Теория вероятностей и математическая статистика; Статистические методы и первичный анализ данных; Введение в базы данных;	Технологическая (проектно-технологическая) практика (производственная); Методы машинного обучения;
ПК-2	Способен проектировать архитектуру информационных систем с компонентами ИИ, разрабатывать прототипы и базы данных таких систем	Технологическая (проектно-технологическая) практика (учебная); Эксплуатационная практика (учебная); Программирование на языке C++; Методы разработки решений на основе искусственного интеллекта (Git, Docker); Алгоритмы и структуры данных; Программирование на языке Python; Введение в базы данных;	Эксплуатационная практика (производственная); Преддипломная практика; Технологическая (проектно-технологическая) практика (производственная); Массово-параллельные вычисления в машинном обучении (GPU); MLOps и промышленная разработка систем искусственного интеллекта; Практическая подготовка на проектах отраслевых индустриальных партнеров; Проектирование и разработка систем компьютерного зрения; Практикум по обработке естественного языка (NLP); Основы глубокого обучения; *Вайб-кодинг***;
BD-3	Способен организовывать хранение данных, выбирая адекватные технологические решения	Введение в базы данных;	Эксплуатационная практика (производственная);
BD-4	Способен применять различные модели и (или) технологии обработки больших данных		Эксплуатационная практика (производственная); Массово-параллельные вычисления в машинном обучении (GPU); MLOps и промышленная разработка систем искусственного интеллекта;
PL-1	Способен применять язык программирования Python для решения задач в области ИИ	Технологическая (проектно-технологическая) практика (учебная); Программирование на языке Python; Алгоритмы и структуры данных;	Технологическая (проектно-технологическая) практика (производственная); Эксплуатационная практика (производственная); *Вайб-кодинг***;

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
		Статистические методы и первичный анализ данных;	Методы машинного обучения; Основы глубокого обучения;

* - заполняется в соответствии с матрицей компетенций и СУП ОП ВО

** - элективные дисциплины /практики

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины «Hadoop, SPARK» составляет «3» зачетные единицы.

Таблица 4.1. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	**ВСЕГО,** ак.ч.		**Семестр(-ы)**
			5
Контактная работа, ак.ч.	*68*		*68*
Лекции (**ЛК**)	17		17
Лабораторные работы (**ЛР**)	0		0
Практические/семинарские занятия (**СЗ**)	51		51
Самостоятельная работа обучающихся, ак.ч.	*13*		*13*
Контроль (экзамен/зачет с оценкой), ак.ч.	*27*		*27*
Общая трудоемкость дисциплины	**ак.ч.**	***108***	*108*
	зач.ед.	***3***	*3*

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5.1. Содержание дисциплины (модуля) по видам учебной работы

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
Раздел 1	Экосистема Hadoop и распределённое хранение данных	1.1	Введение в Big Data. Экосистема Hadoop	Определение Big Data: 5V (Volume, Velocity, Variety, Veracity, Value). Эволюция подходов: от реляционных СУБД к распределённым системам. Экосистема Apache Hadoop: история, компоненты (HDFS, YARN, MapReduce), место в современном ландшафте. Сравнение Hadoop и Spark. Облачные аналоги: Amazon EMR, Google Dataproc, Yandex Data Proc. Связь с задачами ИИ: обучение на больших данных, обработка логов, feature engineering	ЛК	ОПК-2.2, BD-4.1, УК-1.3
		1.2	HDFS и YARN: архитектура и принципы	HDFS: архитектура (NameNode, DataNode), блоки, репликация, rack awareness. Модель записи: write-once-read-many. Отказоустойчивость: репликация блоков, Secondary NameNode, HA NameNode. YARN: архитектура (ResourceManager, NodeManager, ApplicationMaster), распределение ресурсов (контейнеры, vCores, память). Планировщики: FIFO, Capacity, Fair. Связь HDFS с Data Lake	ЛК	BD-4.1, BD-3.1, ОПК-2.2
		1.3	Практикум: развёртывание кластера Hadoop и работа с HDFS	Развёртывание Hadoop-кластера в Docker (docker-compose с NameNode, DataNode, ResourceManager). Команды HDFS CLI: hdfs dfs -ls, -put, -get, -mkdir, -rm, -cat, -du. Загрузка данных в HDFS. Просмотр статуса кластера через веб-интерфейс (NameNode UI, YARN UI). Проверка репликации блоков	СЗ	BD-4.1, ОПК-2.3
		1.4	Практикум: паттерн MapReduce — принцип и реализация	Паттерн MapReduce: Map (преобразование) → Shuffle/Sort → Reduce (агрегация). Реализация MapReduce на Python (Hadoop Streaming): mapper.py, reducer.py. Задача: подсчёт слов в корпусе текстов (WordCount). Запуск на кластере (hadoop jar hadoop-streaming.jar). Анализ логов выполнения, счётчиков	СЗ	BD-4.1, ОПК-2.3
		1.5	Практикум: MapReduce — задачи обработки данных	Реализация MapReduce для прикладных задач: вычисление средней и медианы по группам, top-N элементов, join двух датасетов (reduce-side join). Паттерны: combiner для уменьшения объёма данных на shuffle, secondary sort.	СЗ	BD-4.1, ОПК-2.3

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				Обсуждение ограничений MapReduce: многофазные задачи, итеративные алгоритмы		
		1.6	Практикум: Apache Hive — SQL поверх Hadoop	Hive: архитектура (Metastore, Driver, Compiler). Создание таблиц: внутренние и внешние (EXTERNAL). Типы данных. Партиционирование (PARTITIONED BY). Форматы хранения: TextFile, ORC, Parquet. HiveQL: SELECT, JOIN, GROUP BY, оконные функции. Практика: создание хранилища данных в Hive, загрузка CSV, аналитические запросы	СЗ	BD-3.1, BD-4.1, ОПК-7.2
		1.7	Практикум: Hive — оптимизация и партиционирование	Партиционирование по дате/категории: ускорение запросов. Бакетирование (CLUSTERED BY). Формат Parquet: колоночное хранение, сжатие, predicate pushdown. ORC vs. Parquet: сравнение. EXPLAIN для анализа плана выполнения. Практика: сравнение времени выполнения запроса на TextFile vs. Parquet, с партиционированием и без	СЗ	BD-3.1, BD-4.1, ПК-2.3
		1.8	Практикум: проектирование Data Lake на HDFS	Концепция Data Lake: raw zone, curated zone, production zone. Организация данных в HDFS: структура каталогов, именование, партиционирование по дате. Формат хранения: Parquet для аналитики, JSON для логов. Метаданные: Hive Metastore. Практика: проектирование структуры Data Lake для ML-проекта (сырые данные, обработанные, признаки, модели)	СЗ	BD-3.1, ПК-2.3, ПК-2.1
		1.9	Практикум: мини-проект — ETL-пайплайн на Hadoop	Сквозная задача: загрузка CSV в HDFS → обработка MapReduce (очистка, трансформация) → загрузка в Hive-таблицу (Parquet, партиционирование) → аналитические запросы HiveQL → визуализация результатов. Документирование архитектуры пайплайна. Обсуждение: ограничения Hadoop, мотивация перехода к Spark	СЗ	BD-4.1, ОПК-7.2, ПК-2.1
Раздел 2	Apache Spark: основы и обработка данных	2.1	Архитектура Apache Spark	Spark vs. MapReduce: in-memory processing, DAG execution, lazy evaluation. Архитектура: Driver, Cluster Manager (Standalone, YARN, Kubernetes), Executors, Tasks. RDD (Resilient Distributed Dataset): партиции, трансформации (lazy), действия (eager), lineage (граф происхождения). Приложения Spark: Spark SQL, MLlib, Structured Streaming, GraphX. PySpark: Python API для Spark	ЛК	BD-4.2, ОПК-2.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
		2.2	Spark DataFrame и Spark SQL	DataFrame API: создание, схема (StructType, StructField), чтение/запись (CSV, Parquet, JSON, JDBC). Трансформации: select, filter, withColumn, groupBy, agg, join, orderBy, window. UDF (User Defined Functions). Spark SQL: регистрация временных таблиц, SQL-запросы. Catalyst optimizer: логический и физический план. Tungsten: управление памятью. Сравнение DataFrame и RDD по производительности	ЛК	BD-4.2, ОПК-2.3
		2.3	Практикум: установка и первые шаги с PySpark	Установка PySpark (pip install pyspark или Docker). Создание SparkSession. Чтение CSV в DataFrame. Команды: show, printSchema, describe, count. Запуск в Jupyter Notebook. Spark UI: просмотр джобов, стадий, задач, DAG. Практика: загрузка датасета, первичный анализ (describe, value_counts, null check)	СЗ	BD-4.2, ОПК-2.3
		2.4	Практикум: трансформации DataFrame	Фильтрация (filter, where), выбор столбцов (select), добавление столбцов (withColumn), переименование (withColumnRenamed). Агрегация: groupBy + agg (count, sum, avg, min, max). Условные выражения: when, otherwise. Работа с NULL: isNull, isNotNull, coalesce, na.fill, na.drop. Практика: предобработка большого датасета	СЗ	BD-4.2, ОПК-2.3
		2.5	Практикум: соединения и оконные функции	Типы JOIN: inner, left, right, full, cross, semi, anti. Broadcast join для маленьких таблиц. Оконные функции: Window.partitionBy.orderBy, row_number, rank, lag, lead, кумулятивные суммы. Практика: обогащение данных через JOIN, вычисление скользящих статистик по временным рядам	СЗ	BD-4.2, ОПК-7.2
		2.6	Практикум: Spark SQL и работа с разными форматами	Регистрация DataFrame как временной таблицы (createOrReplaceTempView). Выполнение SQL-запросов (spark.sql). Чтение и запись в разных форматах: CSV, Parquet, JSON, Delta Lake (обзор). Партиционирование при записи (partitionBy). Практика: перевод Hive-запросов из раздела 1 на Spark SQL, сравнение производительности	СЗ	BD-4.2, BD-3.1
		2.7	Практикум: оптимизация Spark-приложений	Catalyst optimizer: EXPLAIN для анализа плана. Основные проблемы: data skew (перекос данных), shuffle, spill. Стратегии: repartition, coalesce, broadcast, cache/persist,	СЗ	BD-4.2, ОПК-2.3

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				salting для борьбы со skew. Настройка параметров: spark.sql.shuffle.partitions, spark.executor.memory, spark.executor.cores. Практика: профилирование и оптимизация медленного запроса		
		2.8	Практикум: UDF и сложные трансформации	Создание UDF (udf) и Pandas UDF (pandas_udf): различия, производительность. Работа со сложными типами: ArrayType, MapType, StructType. Функции: explode, collect_list, collect_set, arrays_zip. Практика: обработка вложенных JSON-данных, извлечение признаков из массивов и структур	СЗ	BD-4.2, ОПК-2.3
		2.9	Практикум: ETL-пайплайн на PySpark	Сквозная задача: чтение сырых данных из нескольких источников (CSV, JSON) → очистка (обработка NULL, выбросов, дубликатов) → трансформация (кодирование, нормализация, feature engineering) → агрегация → запись результатов в Parquet с партиционированием. Документирование пайплайна. Сравнение с Pandas по скорости на большом датасете	СЗ	BD-4.2, ОПК-7.2, ПК-2.3
Раздел 3	Spark MLlib и распределённое машинное обучение	3.1	Spark MLlib: архитектура и пайплайны	Spark MLlib: DataFrame-based API (spark.ml). Основные абстракции: Transformer, Estimator, Pipeline, ParamGrid. Пайплайн: цепочка этапов (StringIndexer → VectorAssembler → StandardScaler → LogisticRegression). Сохранение и загрузка пайплайнов (save/load). Связь с концепцией ML-пайплайнов в scikit-learn	ЛК	BD-4.2, ОПК-2.2
		3.2	Spark MLlib: предобработка и feature engineering	Трансформеры предобработки: StringIndexer, OneHotEncoder, VectorAssembler, StandardScaler, MinMaxScaler, Bucketizer, SQLTransformer. Обработка текста: Tokenizer, StopWordsRemover, HashingTF, IDF. Обработка категорий: StringIndexer + OneHotEncoder. Feature selection: ChiSqSelector. Практика: построение пайплайна предобработки для табличных данных	ЛК	BD-4.2, ОПК-2.3
		3.3	Практикум: классификация на Spark MLlib	Загрузка датасета в Spark DataFrame. Построение пайплайна: предобработка → LogisticRegression. Обучение (fit) и предсказание (transform). Оценка: BinaryClassificationEvaluator (AUC), MulticlassClassificationEvaluator (accuracy, F1).	СЗ	BD-4.2, ОПК-2.3

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				CrossValidator для кросс-валидации. ParamGridBuilder для подбора гиперпараметров. Сравнение с scikit-learn по результатам и времени		
		3.4	Практикум: регрессия и деревья на Spark MLlib	Модели: LinearRegression, DecisionTreeRegressor, RandomForestRegressor, GBTRegressor. Оценка: RegressionEvaluator (RMSE, MAE, R^2). Feature importance для деревьев. Подбор гиперпараметров: CrossValidator + ParamGridBuilder. Практика: прогнозирование цен на большом датасете. Визуализация результатов	СЗ	BD-4.2, ОПК-7.2
		3.5	Практикум: кластеризация и рекомендации на Spark MLlib	Кластеризация: KMeans, BisectingKMeans, GaussianMixture. Оценка: ClusteringEvaluator (silhouette). Рекомендательные системы: ALS (Alternating Least Squares) для коллаборативной фильтрации. Практика: кластеризация клиентов по поведению, построение рекомендательной системы на датасете рейтингов (MovieLens)	СЗ	BD-4.2, ОПК-2.3
		3.6	Практикум: полный ML-пайплайн на Spark	Сквозная задача: чтение данных → EDA (Spark SQL + визуализация) → предобработка (Pipeline) → обучение нескольких моделей (LR, RF, GBT) → кросс-валидация → выбор лучшей модели → сохранение пайплайна. Работа с датасетом, не помещающимся в память одной машины. Документирование: описание пайплайна, метрики, обоснование выбора	СЗ	BD-4.2, ОПК-7.2, ПК-2.1
		3.7	Практикум: интеграция Spark с внешними хранилищами	Чтение из PostgreSQL (JDBC), MongoDB (Spark MongoDB Connector), Parquet на HDFS/S3. Запись результатов обратно. Delta Lake (обзор): ACID-транзакции, time travel, schema enforcement. Практика: чтение обучающих данных из PostgreSQL, обучение модели в Spark, сохранение предсказаний в Parquet/Delta	СЗ	BD-3.1, BD-3.2, ПК-2.3
		3.8	Практикум: Feature Store на Spark (упрощённая версия)	Проектирование распределённого Feature Store: вычисление признаков на Spark, хранение в Parquet/Delta с партиционированием по дате и entity_id. Инкрементальное обновление признаков. Чтение признаков для обучения и инференса. Обсуждение: Feast, Tecton, Hopsworks (обзор). Связь с промышленной практикой MLOps	СЗ	BD-3.1, ПК-2.3, ПК-2.1
Раздел 4	Потоковая обработка	4.1	Потоковая обработка данных: концепции	Пакетная vs. потоковая обработка: batch, micro-batch, true	ЛК	BD-4.3,

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
	данных и архитектура Big Data для ИИ		и Apache Kafka	streaming. Паттерны: Lambda-архитектура (batch + streaming), Kappa-архитектура (only streaming). Apache Kafka: архитектура (брокеры, топики, партиции, consumer groups), гарантии доставки (at-least-once, at-most-once, exactly-once). Kafka как связующее звено в real-time ML-пайплайнах		ОПК-2.2
		4.2	Spark Structured Streaming	Structured Streaming: модель «бесконечной таблицы» (unbounded table). Источники: Kafka, файлы, сокеты. Операции: select, filter, groupBy, window (tumbling, sliding, session). Режимы вывода: append, complete, update. Watermarking для обработки опоздавших данных. Checkpointing для отказоустойчивости. Связь с задачами ИИ: мониторинг дрейфа данных, online feature engineering	ЛК	BD-4.3, BD-4.2
		4.3	Практикум: Apache Kafka — установка и базовые операции	Развёртывание Kafka в Docker (docker-compose: zookeeper + kafka). Создание топика (kafka-topics.sh). Producer: отправка сообщений (kafka-console-producer, confluent-kafka-python). Consumer: чтение сообщений (kafka-console-consumer, confluent-kafka-python). Мониторинг: kafka-consumer-groups, lag	СЗ	BD-4.3, ОПК-2.3
		4.4	Практикум: потоковая обработка с Spark Structured Streaming	Чтение потока из Kafka (readStream, format("kafka")). Парсинг JSON-сообщений. Агрегация в окнах: подсчёт событий за последние 5 минут (window, groupBy). Watermarking. Запись результатов: console sink, Parquet sink, Kafka sink. Практика: подсчёт статистик по потоку событий в реальном времени	СЗ	BD-4.3, BD-4.2
		4.5	Практикум: потоковый мониторинг ML-модели	Задача: мониторинг предсказаний ML-модели в реальном времени. Поток предсказаний записывается в Kafka. Structured Streaming: чтение потока, вычисление скользящих метрик (средний confidence, доля каждого класса, процент аномалий). Обнаружение дрейфа: сравнение текущего распределения с эталонным. Алерт при превышении порога	СЗ	BD-4.3, ОПК-7.2, PL-1.3
		4.6	Практикум: архитектура Big Data для ML-проекта	Проектирование архитектуры данных для ML-проекта: источники (API, логи, БД) → ingestion (Kafka) → storage (HDFS/S3, Parquet/Delta) → processing (Spark batch +	СЗ	ПК-2.1, УК-1.3, ОПК-2.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				streaming) → feature store → model training (Spark MLlib / PyTorch) → serving → monitoring. Составление архитектурной диаграммы. Обоснование выбора технологий для каждого компонента		
		4.7	Практикум: сравнение технологий и выбор стека	Сравнительный анализ: Hadoop MapReduce vs. Spark (производительность, API, экосистема). Hive vs. Spark SQL (синтаксис, производительность, применимость). Spark vs. Dask vs. Ray (масштабирование, API, экосистема ML). Kafka vs. RabbitMQ vs. Redis Streams (гарантии, производительность, применимость). Методология выбора: требования → критерии → сравнение → обоснование	СЗ	УК-1.3, ОПК-2.2, ПК-2.1
		4.8	Практикум: итоговый проект — распределённый ML-пайплайн на Spark	Финальная интеграция: ingestion (Kafka → Spark Streaming) → хранение (HDFS/Parquet) → batch-обработка (Spark SQL, ETL) → feature engineering → обучение модели (Spark MLlib) → мониторинг предсказаний (Streaming). Docker Compose для всех компонентов. Документирование: архитектурная диаграмма, описание компонентов, обоснование выбора технологий, метрики производительности. Презентация	СЗ	BD-4.1, BD-4.2, BD-4.3, ПК-2.1, ПК-2.3

* - заполняется только по **ОЧНОЙ** форме обучения: *ЛК – лекции; ЛР – лабораторные работы; СЗ – практические/семинарские занятия.*

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Таблица 6.1. Материально-техническое обеспечение дисциплины

Тип аудитории	Оснащение аудитории	Специализированное учебное/лабораторное оборудование, ПО и материалы для освоения дисциплины (при необходимости)
Лекционная	Аудитория для проведения занятий лекционного типа, оснащенная комплектом специализированной мебели; доской (экраном) и техническими средствами мультимедиа презентаций.	
Семинарская	Аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная комплектом специализированной мебели и техническими средствами мультимедиа презентаций.	Персональные компьютеры, необходимое ПО
Для самостоятельной работы	Аудитория для самостоятельной работы обучающихся (может использоваться для проведения семинарских занятий и консультаций), оснащенная комплектом специализированной мебели и компьютерами с доступом в ЭИОС.	Персональные компьютеры, необходимое ПО

* - аудитория для самостоятельной работы обучающихся указывается **ОБЯЗАТЕЛЬНО**!

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература:

1. Лэм Чак. Hadoop в действии [Электронный ресурс]. - М.: ДМК Пресс, 2019. 424 с. ISBN 978-5-97060-723-7 URL: https://mega.rudn.ru/MegaPro/UserEntry?Action=Link_FindDoc&id=475248&idb=0

2. Карау Х., Конвински Э., Венделл П., Захария М. Изучаем Spark: молниеносный анализ данных. - М.: ДМК Пресс, 2015. - 304 с.: ил. ISBN 978-5-97060-323-9

Дополнительная литература:

1. Сэнди Риза, Ури Лезерсон, Шон Оуэн, Джош Уиллс. Spark для профессионалов: современные паттерны обработки больших данных. — СПб.: Питер, 2017. — 272 с.: ил. — (Серия «Бестселлеры O’Reilly»). ISBN 978-5-496-02401-3

2. Машнин Т. Технология хранения и обработки больших данных Hadoop / Т. Машнин. — Москва, 2026. — 136 с. — ISBN 978-5-532-96881-3.

Ресурсы информационно-телекоммуникационной сети «Интернет»:

1. ЭБС РУДН и сторонние ЭБС, к которым студенты университета имеют доступ на основании заключенных договоров

- Электронно-библиотечная система РУДН – ЭБС РУДН https://mega.rudn.ru/MegaPro/Web
- ЭБС «Университетская библиотека онлайн» http://www.biblioclub.ru

- ЭБС «Юрайт» http://www.biblio-online.ru
- ЭБС «Консультант студента» www.studentlibrary.ru
- ЭБС «Знаниум» https://znanium.ru/

2. Базы данных и поисковые системы
   - Sage https://journals.sagepub.com/
   - Springer Nature Link https://link.springer.com/
   - Wiley Journal Database https://onlinelibrary.wiley.com/
   - Наукометрическая база данных Lens.org https://www.lens.org

Учебно-методические материалы для самостоятельной работы обучающихся при освоении дисциплины/модуля:*

1. Курс лекций по дисциплине «Hadoop, SPARK».

* - все учебно-методические материалы для самостоятельной работы обучающихся размещаются в соответствии с действующим порядком на странице дисциплины **в ТУИС**!