

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

УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования – первый
проректор

_____ Хагуров Т.А.

«02» июня 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1. О.40 Технологии обработки больших данных

Направление подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем

Профиль Искусственный интеллект и аналитика данных

Форма обучения очная

Квалификация бакалавр

Краснодар 2026

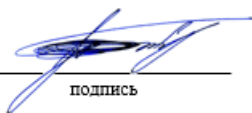
Рабочая программа дисциплины «Технологии обработки больших данных» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем

Программу составил(и):
Е.Н. Калайдин, доктор физ.-мат. наук



подпись

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



подпись

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



подпись

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

Луценко Евгений Вениаминович, доктор экономических наук, кандидат технических наук, профессор кафедры компьютерных технологий и систем Федерального государственного бюджетного образовательного учреждения высшего образования «Кубанский государственный аграрный университет имени И.Т. Трубиллина», e-mail: prof.lutsenko@gmail.com

1 Цели и задачи изучения дисциплины (модуля)

1.1 Цель освоения дисциплины

Цель дисциплины - Изучение принципов обработки больших данных, технологий распределенных вычислений, облачных платформ и инструментов анализа данных.

1.2 Задачи дисциплины

- Изучение архитектурных решений для работы с Big Data.
- Освоение методов обработки структурированных и неструктурированных данных.
- Применение распределенных вычислений (Hadoop, Spark).
- Разработка алгоритмов анализа данных в распределенных средах.
- Использование облачных платформ для обработки больших данных.

Требования к знаниям и навыкам:

- Умение работать с распределенными системами (Hadoop, Spark).
- Опыт обработки данных в облачных средах (Yandex Cloud).
- Навыки анализа данных с помощью Python (Pandas, PySpark, Dask).
- Понимание архитектуры Big Data-решений.

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Технологии обработки больших данных» относится к Блок 1 Дисциплины, Обязательная часть учебного плана.

Входными знаниями (пререквизиты) для освоения данной дисциплины являются знания, умения и опыт, накопленный студентами в процессе изучения дисциплины «Базы данных», «Технологии управления данными в NoSQL», «Машинное обучение», «Обработка данных на Python».

Постреквизиты: DevOps / MLOps, Глубокое обучение, Современные методы фронтальных исследований ИИ, Методы обучения с подкреплением, Технологии обработки языка, звуковых данных, включая распознавание и синтез речи, Генеративный искусственный интеллект.

Смежные: Облачные технологии и бэкэндразработка, Технологии управления данными NoSQL.

1.4 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

Код и содержание компетенции	Общий индикатор	Уровни освоения индикаторов компетенции
1	2	3
ОПК-2 Способен применять современный математический аппарат, связанный с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности	ОПК-2.1 Способен применять системный подход к анализу предметной (проблемной) области, выявлению требований к ИС	Знать: методы системного анализа, этапы выявления требований к информационным системам (ИС), современный математический аппарат для проектирования программных продуктов. Уметь: применять системный подход для анализа предметной области, формализовать требования к ИС, выбирать адекватные математические модели для оценки качества программных продуктов. Владеть: навыками применения математического аппарата для разработки и оценки качества программных продуктов и комплексов.
BD-3 Способен организовывать хранения данных	BD-3.2 Разрабатывает, отлаживает и тестирует прикладные решения с	Знать: основные типы NoSQL-хранилищ (Ключ-Значение, Доку-

Код и содержание компетенции	Общий индикатор	Уровни освоения индикаторов компетенции
1	2	3
выбирая адекватные технологические решения	элементами ИИ с применением различных технологий хранения неструктурированных данных, оценивает качество.	ментные, Колоночные, Графовые); базовые команды для работы с данными в таких хранилищах. Уметь: создавать базы данных в указанных хранилищах, выполнять CRUD-операции, использовать индексы; разрабатывать, отлаживать и тестировать прикладные решения с элементами ИИ. Владеть: технологиями хранения неструктурированных данных на уровне применения наиболее известных подходов каждого класса хранилищ; оценивать качество решений.
BD-4 Способен применять различные модели и (или) технологии обработки данных	BD-4.1 Осуществляет выбор технологий обработки больших данных, приемлемых для создания прикладной системы ИИ с заданными требованиями	Знать: современные технологии распределённого хранения и параллельной обработки (Hadoop, Spark); архитектурные принципы Big Data-решений. Уметь: организовывать распределённое хранилище и параллельную обработку на базе Hadoop/Spark; осуществлять выбор технологий под заданные требования. Владеть: навыками развёртывания и настройки кластеров (в т.ч. в облачных средах), оптимизации вычислительных процессов.
	BD-4.2 Разрабатывает и отлаживает прикладные решения с элементами ИИ с применением технологий обработки данных	Знать: методы и алгоритмы обработки данных в распределённых средах (ETL, агрегация, анализ). Уметь: разрабатывать и отлаживать прикладные решения с элементами ИИ с использованием технологий обработки данных (Spark, SQL, потоковая обработка). Владеть: инструментарием для реализации полных циклов обработки данных (от загрузки до визуализации).
BD- 5 Способен применять технологии организации инфраструктуры БД	BD-5.1 Осуществляет выбор направления вспомогательных технологических решений для формирования единого стека работы с большими данными для решения поставленной задачи	Знать: основные компоненты инфраструктуры для работы с большими данными (хранилища, очереди, оркестрация). Уметь: выполнять отдельные функции в проектах по созданию инфраструктуры БД; выбирать и обосновывать вспомогательные технологические решения. Владеть: практическими навыками настройки, мониторинга и интеграции элементов инфраструктуры (кластеры, облачные сервисы).

Код и содержание компетенции	Общий индикатор	Уровни освоения индикаторов компетенции
1	2	3
	BD-5.2 Разрабатывает и отлаживает прикладные решения с элементами ИИ с применением технологий организации БД и инфраструктуры	Знать: различные технологии организации баз данных и инфраструктуры (NoSQL, объектные хранилища, репликация, шардинг). Уметь: участвовать в разработке решений с элементами ИИ, применяя разные технологии организации инфраструктуры БД. Владеть: методами интеграции прикладных решений с элементами ИИ в существующую инфраструктуру обработки данных.
LC-8 Способен применять знания об истории развития и трендах современного ИИ для формулирования корректных постановок задач и поиска перспективных способов решения проблем с помощью ИИ	LC-8.2 Определяет тенденции развития, оценивает новизну и практическую значимость своих решений с точки зрения современного искусственного интеллекта	Знать: причины появления концепции больших данных, разницу определений; категории проблем больших данных; историю и динамику появления новых технологий. Уметь: выявлять различные категории проблем больших данных с примерами; анализировать динамику появления технологий; сопоставлять собственные решения с современными исследованиями и индустриальными стандартами; определять тенденции развития. Владеть: навыками критической оценки новизны и практической значимости своих решений в контексте современного ИИ.
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	Знать: основные функции фреймворка PySpark, возможности Airflow для оркестрации. Уметь: применять PySpark для обработки данных, использовать Airflow для построения процессов обработки больших данных. Владеть: навыками разработки и поддержки систем обработки больших данных различной степени сложности.

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 3 зач. ед. (108 часов), их распределение по видам работ представлено в таблице

Вид учебной работы	Всего часов	Семестры (часы)
		6
Контактная работа, в том числе:	68,3	68,3
Аудиторные занятия (всего):	64	64
Занятия лекционного типа	32	32
Лабораторные занятия	32	32
Иная контактная работа:	4,3	4,3
Контроль самостоятельной работы (КСР)	4	4
Промежуточная аттестация (ИКР)	0,3	0,3
Самостоятельная работа, в том числе:	4	4
Подготовка к текущему контролю	4	4

Контроль:		35,7	35,7
Подготовка к экзамену		35,7	35,7
Общая трудоемкость	час.	108	108
	в том числе контактная работа	68,3	68,3
	зач. ед	3	3

2.2 Структура дисциплины

№	Наименование разделов (тем)	Количество часов			
		Всего	Аудиторная работа		Внеаудиторная работа
			Л	ЛР	СРС
1	2	3	4	6	7
1.	Введение в Big Data	4	2	2	
2.	Архитектуры обработки больших данных	4	2	2	
3.	Hadoop и экосистема	4	2	2	
4.	Apache Spark	4	2	2	
5.	Обработка потоковых данных	6	4	2	
6.	NoSQL для больших данных	4	2	2	
7.	Облачные технологии для Big Data	6	4	2	
8.	Методы анализа больших данных	6	2	4	
9.	Машинное обучение на больших данных	6	2	4	
10.	Визуализация больших данных	4	2	2	
11.	Оптимизация и масштабирование	4	2	2	
12.	Безопасность данных	6	2	2	2
13.	Реальные кейсы использования Big Data	6	2	2	2
14.	Этика и законодательство в Big Data	4	2	2	
ИТОГО по разделам дисциплины		68	32	32	4
Контроль самостоятельной работы (КСР)		35,7			
Промежуточная аттестация (ИКР)		0,3			
Общая трудоемкость по дисциплине		108			

2.3 Содержание разделов (тем) дисциплины

2.3.1 Занятия лекционного типа

№	Наименование раздела (темы)	Содержание раздела (темы)	Соответствие индикаторам компетенций
1	2	3	
1	Введение в Big Data	Понятие больших данных (Volume, Velocity, Variety, Veracity, Value). Области применения и примеры использования (социальные сети, IoT, финансы, медицина). Различия между традиционными СУБД и Big Data-решениями.	ОПК-2.1 – применение системного подхода к анализу проблемной области больших данных; LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость своих решений с точки зрения современного ИИ
2	Архитектура распределенных систем	Принципы распределенных вычислений. Модели распределенных вычислений. Hadoop, Spark, Kafka	ОПК-2.1 – системный подход к выбору архитектуры ИС на основе требований; BD-4.1 – осуществляет выбор технологий обработки больших данных для создания системы ИИ; PL-1.3 –

№	Наименование раздела (темы)	Содержание раздела (темы)	Соответствие индикаторам компетенций
1	2	3	
			разрабатывает и поддерживает системы обработки больших данных различной сложности
3	Экосистема Hadoop	Архитектура Hadoop (HDFS, YARN). Инструменты (Hive, Pig, HBase). Модель вычислений MapReduce. Аналоги: Apache Spark, преимущества и недостатки.	BD-4.1 – осуществляет выбор технологий обработки больших данных; BD-5.1 – осуществляет выбор вспомогательных технологических решений для единого стека работы с Big Data
4	Обработка данных в Apache Spark	RDD, DataFrame, Dataset. Оптимизация вычислений. Spark SQL и оптимизация запросов.	BD-3.2 – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ с применением технологий хранения неструктурированных данных; BD-4.1 – осуществляет выбор технологий обработки больших данных
5	Обработка потоковых данных	Apache Kafka: архитектура и применение. Apache Flink / Spark Streaming. Обработка событий в реальном времени.	BD-4.1 – осуществляет выбор технологий обработки больших данных; PL-1.3 – разрабатывает и поддерживает системы обработки больших данных различной сложности
6	NoSQL для Big Data	Типы NoSQL (документоориентированные, ключ-значение, графовые, колоночные). Cassandra, MongoDB, Neo4j в контексте Big Data.	BD-5.1 – осуществляет выбор вспомогательных технологических решений для формирования единого стека работы с большими данными
7	Облачные технологии для Big Data	Yandex Data Proc, Yandex Object Storage. Развертывание кластера	BD-4.1 – осуществляет выбор технологий обработки больших данных; BD-5.1 – осуществляет выбор вспомогательных технологических решений
8	Методы анализа больших данных	Классификация, кластеризация, рекомендательные системы	ОПК-2.1 – выявление требований к методам анализа в зависимости от предметной области; BD-4.1 – осуществляет выбор технологий обработки больших данных; LC-8.2 – определяет тенденции развития, оценивает новизну

№	Наименование раздела (темы)	Содержание раздела (темы)	Соответствие индикаторам компетенций
1	2	3	
9	Машинное обучение на больших данных	Особенности обучения моделей в распределённых средах. Feature Engineering для Big Data. MLlib в Spark. Применение TensorFlow/PyTorch в Spark.	BD-4.1 – осуществляет выбор технологий обработки больших данных; LC-8.2 – определяет тенденции развития, оценивает новизну
10	Визуализация и мониторинг Big Data	Инструменты (Grafana, Kibana, Tableau). Dash, Plotly, Superset. Логирование и отладка распределённых приложений.	(без привязки к удалённым компетенциям)
11	Оптимизация и масштабирование	Шардинг, репликация, кэширование	BD-4.1 – осуществляет выбор технологий обработки больших данных; BD-5.1 – осуществляет выбор вспомогательных технологических решений
12	Безопасность данных	Аутентификация, авторизация, шифрование.	BD-5.1 – осуществляет выбор вспомогательных технологических решений для формирования единого стека работы с большими данными
13	Кейсы и тренды в Big Data	Анализ логов, рекомендательные системы, IoT. Облачные решения (AWS EMR, Google Dataproc).	ОПК-2.1 – системный анализ реальных кейсов для выявления требований к ИИ-решениям; BD-4.1 – осуществляет выбор технологий обработки больших данных; LC-8.2 – определяет тенденции развития, оценивает новизну
14	Этика и законодательство в Big Data	GDPR, персональные данные	LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость решений

2.3.2 Лабораторные занятия

№	Тема	Задание	Соответствие индикаторам компетенций
1	Настройка окружения (Colab + Yandex Cloud)	Развернуть виртуальную машину в Yandex Cloud, подключиться через Jupyter.	ОПК-2.1 – анализ требований к инфраструктуре и выбор конфигурации; BD-4.1 – выбор технологий обработки больших данных; BD-5.1 – выбор вспомогательных технологических решений
2	Основы PySpark	Загрузка данных в RDD, простые операции (map, filter, reduce).	BD-3.2 – разработка, отладка, тестирование прикладных решений с элементами ИИ; BD-4.1 – выбор

			технологий обработки больших данных
3	Работа с Spark DataFrame	Анализ CSV-файла (агрегации, фильтрация).	BD-3.2 – разработка, отладка, тестирование; BD-4.1 – выбор технологий обработки больших данных
4	Хранение данных в Yandex Object Storage	Загрузка и выгрузка данных из S3-совместимого хранилища.	BD-5.1 – выбор вспомогательных технологических решений
5	Парсинг и обработка логов	Анализ веб-логов с помощью Spark.	ОПК-2.1 – выявление требований к обработке логов на основе анализа предметной области; BD-3.2 – разработка, отладка, тестирование; LC-8.2 – определение тенденций развития, оценка новизны
6	Работа с NoSQL (Yandex Managed MongoDB)	CRUD-операции, индексы.	BD-5.1 – выбор вспомогательных технологических решений
7	Потоковая обработка (Kafka + Spark Streaming)	Чтение данных из Kafka-топика.	BD-4.1 – выбор технологий обработки больших данных; PL-1.3 – разработка и поддержка систем обработки больших данных
8	Машинное обучение в Spark (MLlib)	Классификация данных.	BD-4.1 – выбор технологий обработки больших данных; LC-8.2 – определение тенденций развития, оценка новизны
9	Визуализация данных (Dash/Plotly)	Построение дашборда.	(без привязки к удаленным компетенциям)
10	Оптимизация запросов в Spark	Кэширование, партиционирование.	BD-3.2 – разработка, отладка, тестирование прикладных решений с элементами ИИ
11	Анализ социального графа (GraphX)	Поиск связей между узлами.	BD-4.1 – выбор технологий обработки больших данных; LC-8.2 – определение тенденций развития
12	Развертывание кластера (Yandex Data Proc)	Запуск Spark-задания в облаке.	BD-4.1 – выбор технологий обработки больших данных; BD-5.1 – выбор вспомогательных технологических решений
13	Анализ временных рядов	Прогнозирование с помощью ARIMA.	BD-4.1 – выбор технологий обработки больших данных; LC-

			8.2 – определение тенденций развития
14	Облачная аналитика (Yandex Metrica API)	Выгрузка и анализ данных.	BD-4.1 – выбор технологий обработки больших данных; BD-5.1 – выбор вспомогательных технологических решений
15	Интеграция с API (Yandex SpeechKit)	Обработка аудио через облачный API.	BD-4.1 – выбор технологий обработки больших данных
16	Финальный проект	Полный анализ датасета (от ETL до визуализации).	ОПК-2.1 – применение системного подхода к анализу предметной области и выявлению требований к ИС на всех этапах проекта; BD-3.2 – разработка, отладка, тестирование; BD-4.1 – выбор технологий обработки больших данных; LC-8.2 – определение тенденций развития

2.3.4 Примерная тематика курсовых работ (проектов)

Не предусмотрены учебным планом

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Изучение теоретического материала	Методические указания по организации самостоятельной работы студентов, утвержденные кафедрой информационных технологий, протокол №1 от 30.08.2019
2	Решение задач	Методические указания по организации самостоятельной работы студентов, утвержденные кафедрой информационных технологий, протокол №1 от 30.08.2019

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,
- в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии

В соответствии с требованиями ФГОС в программа дисциплины предусматривает использование в учебном процессе следующих образовательные технологии: чтение лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

- Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.
- Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.
- Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.
- Информационно-коммуникационные технологии (ИКТ) - расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:
 - Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.
 - Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.
 - Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.
 - Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.
 - Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.
 - Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.
 - Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеназванных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Основные виды интерактивных образовательных технологий включают в себя:

- работа в малых группах (команде) - совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;
- проектная технология - индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;
- анализ конкретных ситуаций - анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;
- развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

Подход разбора конкретных задач и ситуаций широко используется как преподавателем, так и студентами во время лекций, лабораторных занятий и анализа результатов самостоятельной работы. Это обусловлено тем, что при исследовании и решении каждой конкретной задачи имеется, как правило, несколько методов, а это требует разбора и оценки целой совокупности конкретных ситуаций.

Темы, задания и вопросы для самостоятельной работы призваны сформировать навыки поиска информации, умения самостоятельно расширять и углублять знания, полученные в ходе лекционных и практических занятий.

Подход разбора конкретных ситуаций широко используется как преподавателем, так и студентами при проведении анализа результатов самостоятельной работы.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4. Оценочные и методические материалы

4.1 Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «название дисциплины».

Оценочные средства включает контрольные материалы для проведения **текущего контроля** в форме отчетов по лабораторным работам и **промежуточной аттестации** в форме вопросов и заданий к экзамену.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

- при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;
- при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;
- при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине (модулю) предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

Структура оценочных средств для текущей и промежуточной аттестации

№ п/п	Код контролируемой компетенции (соответствие индикатору)	Контролируемые разделы		
		Темы лекций	Текущий контроль	Промежуточная аттестация
			Задания лабораторных работ	Экзаменационные вопросы
1	ОПК-2.1 – способен применять системный подход к анализу предметной (проблемной) области, выявлению требований к ИС	1,2,8,13	1,5,8,13,16	1,2,23,24
2	BD-3.2 – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ с применением различных технологий хранения неструктурированных данных, оценивает качество	3,4,6	2,3,5,10,16	2,3,5,6,8
3	BD-4.1 – осуществляет выбор технологий обработки больших данных, приемлемых для создания прикладной системы ИИ с заданными требованиями	2,3,4,7,11	1,2,3,7,11,12,13,14,15,16	1,2,3,4,5,6,21
4	BD-4.2 – разрабатывает и отлаживает прикладные решения с элементами ИИ с применением технологий обработки данных	2,3,4,5,7	7,8,13,16	4,7,8,21
5	BD-5.1 – осуществляет выбор направления вспомогательных технологических решений для формирования единого стека работы с большими данными для решения поставленной задачи	3,6,7,12	1,4,6,12,14	9,10,11,12,25

6	BD-5.2 – разрабатывает и отлаживает прикладные решения с элементами ИИ с применением технологий организации БД и инфраструктуры	3,6,7,12	6,11,12,14	9,10,11,12,25
7	LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость своих решений с точки зрения современного искусственного интеллекта	1,8,9,10,13,14	5,8,9,11,13,16	1,13,14,20,24
8	PL-1.3 – разрабатывает и поддерживает системы обработки больших данных различной степени сложности	2,5	7	7,8,21,22

Показатели, критерии и шкала оценки сформированных компетенций

ОПК-2.1 – способен применять системный подход к анализу предметной (проблемной) области, выявлению требований к ИС

- **Пороговый:** Знает основные понятия системного анализа, этапы выявления требований; может перечислить базовые математические методы, используемые при проектировании ПО.
- **Базовый:** Применяет системный подход для декомпозиции предметной области; составляет формальные спецификации требований; использует стандартные математические модели (например, UML, IDEF0) для описания ИС.
- **Продвинутый:** Выполняет комплексный анализ проблемной области с выделением противоречий; разрабатывает требования к ИС с учетом всех заинтересованных сторон; применяет современный математический аппарат (теория массового обслуживания, теория графов, метрики качества) для оценки и оптимизации программных продуктов.

BD-3.2 – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ с применением различных технологий хранения неструктурированных данных, оценивает качество

- **Пороговый:** Знает базовый синтаксис SQL и Spark SQL; может написать простые запросы (SELECT, WHERE, GROUP BY); понимает разницу между RDD и DataFrame; выполняет CRUD-операции в одном типе NoSQL.
- **Базовый:** Пишет сложные запросы с JOIN, подзапросами, агрегациями; использует оптимизацию (кэширование, партиционирование); понимает план выполнения запроса в Spark; работает с документными и колоночными хранилищами.
- **Продвинутый:** Анализирует и оптимизирует дорогостоящие запросы; применяет оконные функции, UDF, сложные трансформации; объясняет различия между Spark SQL и Hive; оценивает качество решений с помощью метрик (точность, полнота, время отклика).

BD-4.1 – осуществляет выбор технологий обработки больших данных, приемлемых для создания прикладной системы ИИ с заданными требованиями

- **Пороговый:** Знает основные компоненты Hadoop (HDFS, YARN); понимает принцип MapReduce; может запустить простой Spark-скрипт; называет области применения Hadoop и Spark.

- **Базовый:** Объясняет архитектуру Spark (Driver, Executor); использует RDD/DataFrame для обработки данных; понимает разницу между Hadoop и Spark; обосновывает выбор технологии под типовую задачу (пакетная vs потоковая).

- **Продвинутый:** Оптимизирует Spark-приложения (настройка памяти, партиций); разбирается в механизме шардинга и репликации; может настроить кластер в облаке; выполняет сравнительный анализ технологий (Flink vs Spark, Hadoop vs Ray) для конкретных требований к задержке и производительности.

BD-4.2 – разрабатывает и отлаживает прикладные решения с элементами ИИ с применением технологий обработки данных

- **Пороговый:** Пишет простые ETL-скрипты на PySpark (загрузка, фильтрация, сохранение); использует готовые алгоритмы MLlib (например, линейная регрессия) без глубокой настройки.

- **Базовый:** Разрабатывает полный пайплайн обработки (очистка, трансформация, feature engineering); настраивает гиперпараметры модели; выполняет валидацию на тестовой выборке.

- **Продвинутый:** Реализует комплексные решения с элементами ИИ (рекомендательные системы, обнаружение аномалий, графовая аналитика) в распределённой среде; отлаживает производительность; интегрирует модели в продуктивную среду (Streaming, API).

BD-5.1 – осуществляет выбор направления вспомогательных технологических решений для формирования единого стека работы с большими данными для решения поставленной задачи

- **Пороговый:** Различает типы NoSQL (документные, графовые, колоночные, ключ-значение); может выполнить CRUD в MongoDB; знает назначение Kafka и Airflow.

- **Базовый:** Настраивает репликацию и шардинг в NoSQL; сравнивает Cassandra и HBase; выбирает очередь сообщений (Kafka vs RabbitMQ) для потоковой задачи; разворачивает простой кластер в облаке.

- **Продвинутый:** Оптимизирует запросы в колоночных СУБД; проектирует единый технологический стек (хранилище + потоковая обработка + оркестрация) под заданные требования (объём, скорость, консистентность); обосновывает выбор вспомогательных инструментов (мониторинг, логирование, безопасность).

BD-5.2 – разрабатывает и отлаживает прикладные решения с элементами ИИ с применением различных технологий организации БД и инфраструктуры

- **Пороговый:** Использует один тип NoSQL для хранения данных приложения; подключается к облачной базе данных из Python/Spark.

- **Базовый:** Интегрирует приложение с несколькими источниками данных (например, MongoDB + S3 + Kafka); настраивает подключение к кластеру через JDBC/ODBC; изменяет кэширование (Redis).

- **Продвинутый:** Разрабатывает решение с элементами ИИ, использующее гетерогенную инфраструктуру (объектное хранилище + графовая БД + потоковые данные); обеспечивает отказоустойчивость и мониторинг; участвует в развёртывании на кластере Kubernetes.

LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость своих решений с точки зрения современного искусственного интеллекта

- **Пороговый:** Объясняет характеристики Big Data (3V и 5V), приводит примеры использования; знает основные этапы развития технологий (от MapReduce до Spark/Flink).

- **Базовый:** Сравнивает технологии, анализирует преимущества и недостатки; сопоставляет собственное решение с индустриальными стандартами; определяет актуальные тренды (Data Lakehouse, MLOps, Real-time ML).
- **Продвинутый:** Критически оценивает выбор технологий для конкретных кейсов; аргументирует новизну и практическую значимость своего решения (публикации, патент, внедрение); прогнозирует развитие Big Data-экосистемы.

PL-1.3 – разрабатывает и поддерживает системы обработки больших данных различной степени сложности

- **Пороговый:** Знает основные функции фреймворка PySpark, понимает концепцию DAG; может написать простой скрипт обработки логов; запускает задание в локальном режиме.
- **Базовый:** Использует Airflow для оркестрации Spark-задач; оптимизирует пайплайн (партиционирование, широкие зависимости); масштабирует решение на кластере из нескольких узлов.
- **Продвинутый:** Проектирует и сопровождает высоконагруженную систему обработки больших данных (ETL на Spark Streaming + Kafka + хранилище); выбирает оптимальные инструменты (Dask, Ray, Flink) для специфических задач; обеспечивает воспроизводимость и мониторинг.

Пример вопроса: «Как Spark обрабатывает данные в памяти? В чём преимущество перед Hadoop?»

BD 5.1: Инфраструктура БД

Пороговый:

Различает типы NoSQL (документные, графовые).
Может выполнить CRUD в MongoDB.

Базовый:

Настраивает репликацию и шардинг.
Сравнивает Cassandra и HBase.

Продвинутый:

Оптимизирует запросы в колоночных СУБД.
Разворачивает кластер в облаке.

Пример вопроса: «Как работает шардинг в MongoDB?»

LC-8.2 (концепции Big Data):

Пороговый:

Объясняет характеристики Big Data, приводит примеры использования.

Базовый:

Сравнивает технологии, анализа преимуществ/недостатков.

Продвинутый:

Критически оценивает выбор технологий для кейсов.

PL-1.3 (инструменты вычислений):

Пороговый:

Знает основы Dask/Ray, и способен писать простые скрипты.

Базовый:

Использует для параллельной обработки, способен сравнить со Spark.

Продвинутый:

Способен выбрать оптимальный инструмент с обоснованием.

Типовая лабораторная работа иллюстрирующая материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Пример.

Лабораторная работа "Анализ социального графа (GraphX)"

Тема: Поиск связей между узлами в графах с использованием Apache Spark GraphX.

Проверяемые индикаторы:

- **BD-4.1** – осуществляет выбор технологий обработки больших данных (обоснование использования GraphX вместо классических алгоритмов в одной машине).
- **BD-4.2** – разрабатывает и отлаживает прикладное решение с элементами ИИ (выявление сообществ, PageRank).
- **BD-5.2** – применяет технологии организации БД (загрузка графа из MongoDB или Neo4j).
- **PL-1.3** – разрабатывает систему обработки графовых данных большого объёма.
- **LC-8.2** – оценивает практическую значимость графового анализа (например, для рекомендаций или поиска мошеннических групп).

Критерии оценки выполнения лабораторной работы

1. Пороговый уровень (оценка 3)

- Загружен граф в GraphX (вершины и рёбра).
- Выполнен простой анализ (подсчёт вершин/рёбер, степени узлов).
- Найдены непосредственные связи между узлами (например, друзья пользователя).

2. Базовый уровень (оценка 4)

- Применён алгоритм PageRank или Connected Components.
- Визуализированы результаты (например, топ-5 узлов по centrality).
- Оптимизированы запросы (кеширование, партиционирование графа).

3. Продвинутый уровень (оценка 5)

- Реализован сложный анализ (выявление сообществ, предсказание связей).
- Интеграция с внешними данными (например, загрузка графа из MongoDB/Neo4j).
- Сравнение производительности GraphX и других графовых инструментов.
- Интерпретация результатов – как выявленные связи можно использовать в рекомендательной системе или Anti-Fraud.

Пример задания и проверяемые индикаторы

Задание:

- Загрузить граф социальной сети (например, данные из VK API или датасета LiveJournal).
- Найти всех друзей пользователя (1 уровень связей).

- Применить PageRank для определения влиятельных узлов.
- Визуализировать граф (например, с помощью NetworkX или Gephi).

Проверяемые индикаторы:

- BD 4.1 (работа с GraphX).
- BD 5.1 (Оптимизация запросов, развертывание кластеров).

Вывод

Лабораторная работа «Анализ социального графа» в первую очередь проверяет:

- BD 4.1 (работа с распределёнными графами в Spark).

Дополнительно могут затрагиваться BD 5.1 (инфраструктура, нестандартные данные).

Для максимального балла студент должен показать не только техническое выполнение, но и интерпретацию результатов (например, как выявленные связи можно использовать в рекомендательной системе).

Экзаменационные материалы для промежуточной аттестации (экзамен).

Вопросы для подготовки к экзамену

1. Что такое Big Data? Основные характеристики (3V).
2. Сравнение Hadoop и Spark.
3. Принципы работы HDFS.
4. Архитектура MapReduce.
5. Spark RDD vs DataFrame.
6. Какие существуют методы оптимизации в Spark?
7. Особенности потоковой обработки данных.
8. Разница между Kafka и RabbitMQ.
9. Типы NoSQL баз данных.
10. Особенности колоночных СУБД (Cassandra).
11. Как работает шардинг в распределенных БД?
12. Облачные решения для Big Data (AWS, GCP).
13. Методы машинного обучения для больших данных.
14. Как работает рекомендательная система на основе коллаборативной фильтрации?
15. Инструменты визуализации больших данных.
16. Методы обеспечения безопасности в Big Data.
17. Что такое GDPR и как он влияет на обработку данных?
18. Применение Big Data в финансах.
19. Как работает алгоритм PageRank?
20. Проблемы этики при работе с персональными данными.
21. Разница между Lambda и Карра архитектурами.
22. Как работает Elasticsearch?
23. Методы борьбы с перекосом данных (skewness).
24. Что такое Data Lake?
25. Как устроена графовая БД (Neo4j)?

Примерные практические задания (к экзамену)

1. Написать MapReduce-программу для подсчета слов.

2. Создать DataFrame в Spark и выполнить агрегацию.
3. Настроить Kafka producer и consumer.
4. Написать SQL-запрос в Hive.
5. Построить гистограмму распределения данных.
6. Обучить модель классификации в MLlib.
7. Загрузить данные в Google BigQuery.
8. Оптимизировать Spark-запрос.
9. Построить граф в Neo4j.
10. Настроить мониторинг логов в Kibana.

Перечень компетенций (части компетенции), проверяемых оценочным средством

- **ОПК-2.1** – способен применять системный подход к анализу предметной (проблемной) области, выявлению требований к ИС;
- **BD-3.2** – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ с применением различных технологий хранения неструктурированных данных;
- **BD-4.1** – осуществляет выбор технологий обработки больших данных для создания системы ИИ;
- **BD-4.2** – разрабатывает и отлаживает прикладные решения с элементами ИИ с применением технологий обработки данных;
- **BD-5.1** – осуществляет выбор вспомогательных технологических решений для единого стека работы с большими данными;
- **BD-5.2** – разрабатывает и отлаживает прикладные решения с элементами ИИ с применением различных технологий организации БД и инфраструктуры;
- **LC-8.2** – определяет тенденции развития, оценивает новизну и практическую значимость своих решений;
- **PL-1.3** – разрабатывает и поддерживает системы обработки больших данных различной степени сложности.

Структура оценочных средств для текущей и промежуточной аттестации).

4.2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Методические рекомендации, определяющие процедуры оценивания лабораторной работы.

Текущая аттестация проводится по лабораторным работам, и может принести в копилку максимум **40 баллов**. В соответствии с критериями оценки выполнения лабораторных работ Пороговый уровень 10 баллов, базовый 20 баллов, продвинутый уровень 40 баллов. Текущий бал определяется усреднением баллов по всем (16) лабораторным работам и как результат будет принадлежать отрезку от 0 до 40 баллов.

Промежуточной аттестацией по дисциплине «Технологии обработки больших данных» является экзамен. Максимальная оценка, которую можно получить в качестве оценки экзамена 60 баллов.

Методические рекомендации, определяющие процедуры оценивания на экзамене.

В экзаменационном билете два теоретических вопроса и одно практическое задание. Каждый раздел билета оценивается в 20 баллов.

Пример оценки ответа на один вопрос:

В билете вопрос №2 из списка экзаменационных вопросов. Этому вопросу соответствуют два индикатора компетенций, например BD-3.2 и BD-4.1.

- По индикатору BD-3.2 достигнут пороговый уровень (10 баллов).
- По индикатору BD-4.1 достигнут продвинутый уровень (20 баллов).

Тогда ответ данного вопроса в билете будет оценён в 15 баллов (среднее арифметическое). Аналогично второй вопрос.

По теоретическому материалу балл определяется усреднением уровня усвоения компетенций по индикаторам соответствующего раздела.

Практическое задание оценивается следующим образом:

- 18-20 баллов: Код работает корректно, использованы оптимальные методы, решение эффективно.
- 11-17 баллов: Код работает, но есть недочёты в оптимизации.
- 1-10 балла: Код требует доработок, но логика верна.
- 0: Код нерабочий или решение неверное.

Экзаменационная оценка в баллах формируется простым суммированием оценок (баллов) за разделы экзамена и оценки (баллы) текущей аттестации за работу в семестре (лабораторные работы).

В стандартной форме экзаменационная оценка определяется следующим соответствием:

- 0 – 49 баллов «неудовлетворительно»;
- 50 – 70 баллов «удовлетворительно»;
- 71 – 85 баллов «хорошо»;
- 86 – 100 баллов «отлично».

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4.3 Методические указания по организации лабораторных работ по дисциплине "Технологии обработки больших данных"

1. Общие сведения

Образовательная программа: «Искусственный интеллект и аналитика данных» Дисциплина "Технологии обработки больших данных".

Вид обеспечения: Проведение лабораторных работ.

Условия применения:

Для успешного выполнения лабораторных работ требуется:

Программное обеспечение:

Python (PySpark, Pandas, Dash/Plotly, MLlib, Kafka, GraphX)

Jupyter Notebook / Google Colab

Apache Spark, Hadoop (Yandex Data Proc)

MongoDB, Kafka

Yandex Cloud CLI и SDK

Аппаратное обеспечение:

Доступ к облачным ресурсам (Yandex Cloud)

Достаточные вычислительные мощности для обработки данных (виртуальные машины, кластеры)

Облачная инфраструктура:

Yandex Cloud (Compute Cloud, Object Storage, Managed MongoDB, Yandex Data Proc, Yandex Metrica API, SpeechKit)

Доступ к S3-совместимому хранилищу.

2. Цели, задачи и ожидаемые результаты

Цель:

- Закрепить теоретические знания на практике.
- Отработать навыки работы с современными инструментами обработки больших данных (Spark, NoSQL, потоковая обработка, ML).
- Обеспечить понимание облачных технологий и распределенных вычислений.
- Подготовить студентов к решению реальных задач в индустрии (ETL, аналитика, визуализация).

Задачи:

1. Организация инфраструктуры:

Настройка облачного окружения (Yandex Cloud, Colab).

Работа с виртуальными машинами, кластерами (Data Proc).

2. Обработка данных:

Освоение PySpark (RDD, DataFrame).

Работа с NoSQL (MongoDB).

Потоковая обработка (Kafka + Spark Streaming).

3. Аналитика и ML:

Применение MLlib для классификации.

Прогнозирование временных рядов (ARIMA).

4. Визуализация и интеграция:

Построение дашбордов (Dash/Plotly).

Работа с API (Yandex Metrica, SpeechKit).

5. Оптимизация и масштабирование:

Кэширование, партиционирование в Spark.

Развертывание кластера (Yandex Data Proc).

Ожидаемые результаты:

После выполнения лабораторных работ студенты смогут:

- Самостоятельно настраивать облачную инфраструктуру для обработки данных.
- Применять Spark для ETL, агрегации и анализа больших данных.

- Работать с NoSQL и потоковыми данными.
- Строить ML-модели и визуализировать результаты.
- Оптимизировать запросы и развертывать распределенные системы.
- Интегрироваться с внешними API (Yandex Cloud, SpeechKit).

3. Порядок реализации

Порядок реализации представлен в проекции на темы и задания Лабораторных работ.

№	Тема лабораторной работы	Задания
1	Установка и настройка Hadoop (HDFS + YARN)	Развернуть кластер Hadoop, проверить работу HDFS и YARN.
2	Основы MapReduce (Python/Java)	Написать и запустить MapReduce-задачу для подсчета слов.
3	Работа с HBase/Cassandra	Создать таблицу, выполнить CRUD-операции.
4	ETL-процесс в Apache Spark (PySpark)	Загрузить данные, выполнить трансформации, сохранить результат.
5	Потоковая обработка (Kafka + Spark Streaming)	Настроить Kafka, обработать поток данных в Spark.
6	Использование Spark SQL	Выполнить SQL-запросы к данным в Spark.
7	Облачные Big Data-сервисы (AWS/GCP)	Развернуть кластер в облаке, загрузить данные.
8	Оптимизация производительности Spark	Применить кэширование, партиционирование.
9	Data Lake (Delta Lake)	Реализовать ACID-транзакции в Delta Lake.
10	Машинное обучение в Spark (MLlib)	Обучить модель классификации/регрессии.
11	Графовая аналитика (Neo4j/GraphX)	Построить граф, выполнить анализ.
12	NLP на больших текстах	Обработать текст с помощью TF-IDF/Word2Vec.
13	Анализ временных рядов	Построить прогноз с помощью ARIMA/Prophet.
14	Data Quality (Great Expectations)	Проверить качество данных с помощью тестов.
15	Безопасность (Apache Ranger)	Настроить RBAC-политики доступа.
16	Реальный кейс: Анализ данных Uber/Netflix	Проанализировать датасет, визуализировать выводы.
17	Финальный проект	Реализовать комплексный Big Data-пайплайн.

3.1. Задача №1: Организация инфраструктуры

Цель: Настроить облачное окружение и освоить работу с виртуальными машинами и кластерами.

Порядок выполнения:

1. Регистрация в Yandex Cloud:

- Создать аккаунт в Yandex Cloud.
- Активировать бесплатный период или образовательный доступ.

2. Создание виртуальной машины (Compute Cloud):

- Выбрать конфигурацию (CPU, RAM, диск).
- Установить ОС (Ubuntu/Debian).
- Настроить SSH-доступ.

3. Подключение к Jupyter Notebook:

- Установить Jupyter Lab на ВМ.
- Настроить доступ через браузер (проброс портов или облачный балансировщик).

4. Работа с Yandex Data Proc (Spark-кластер):

- Создать кластер Hadoop/Spark.

- Подключиться через Zeppelin или Jupyter.

5. Интеграция с Google Colab:

- Настроить подключение к Yandex Cloud API.
- Загрузить данные из Object Storage в Colab.

6. Оформление отчета по настройке окружения в соответствии с академическими стандартами

3.2. Задача №2: Обработка данных

Цель: Освоить PySpark, NoSQL и потоковую обработку.

Порядок выполнения:

1. Основы PySpark (RDD):

- Загрузить данные в RDD (текст, CSV).
- Применить map, filter, reduce.
- Реализовать WordCount.

2. Spark DataFrame:

- Загрузить CSV/JSON в DataFrame.
- Выполнить агрегации (groupBy, agg).
- Оптимизировать запросы через explain().

3. NoSQL (Yandex Managed MongoDB):

- Создать базу данных в Yandex Cloud.
- Выполнить CRUD-операции (вставка, выборка, обновление).
- Протестировать индексы.

4. Потоковая обработка (Kafka + Spark Streaming):

- Развернуть Kafka-топик в Yandex Cloud.
- Настроить Spark Streaming для чтения данных.
- Визуализировать поток в реальном времени.

5. Сформулировать гипотезу о распределении данных и проверить её с помощью статистических методов

3.3. Задача №3: Аналитика и ML

Цель: Применить MLlib и методы прогнозирования.

Порядок выполнения:

1. Классификация (MLlib):

- Загрузить датасет (например, Iris).
- Разделить данные на train/test.
- Обучить модель (логистическая регрессия/дерево решений).
- Оценить точность.

2. Прогнозирование временных рядов (ARIMA):

- Загрузить временной ряд (например, продажи).
- Построить ARIMA-модель.
- Спрогнозировать значения на будущее.

3.4. Задача №4: Визуализация и интеграция

Цель: Научиться строить дашборды и работать с API.

Порядок выполнения:

1. Дашборды (Dash/Plotly):

- Создать интерактивный график (гистограмма, scatter-plot).
- Добавить фильтры (Dropdown, Slider).

- Развернуть дашборд в облаке.
2. **Работа с API (Yandex Metrica/SpeechKit):**

- Получить API-ключ для Yandex Metrica.
- Загрузить данные посещаемости сайта.
- Обработать аудио через SpeechKit (ASR).

3.5. Задача №5: Оптимизация и масштабирование

Цель: Научиться ускорять Spark-запросы и управлять кластерами.

Порядок выполнения:

1. **Оптимизация Spark:**

- Применить `cache()` / `persist()`.
- Использовать партиционирование данных.
- Сравнить скорость выполнения до/после оптимизации.

2. **Развертывание кластера (Yandex Data Proc):**

- Запустить Spark-задание через Yandex CLI.
- Мониторить ресурсы (CPU, RAM) в Yandex Cloud.

4. Порядок проверки корректности

Каждая задача разбита на четкие шаги, что позволяет студентам последовательно осваивать технологии.

Критерии успешного выполнения:

1. Корректная настройка инфраструктуры.
2. Умение применять Spark, NoSQL, Kafka.
3. Построение ML-моделей и дашбордов.
4. Оптимизация и масштабирование решений.

Чек-лист для проверки выполнения задач по лабораторным работам

Задача №1: Организация инфраструктуры

Действия:

- Создан аккаунт в Yandex Cloud.
- Активирован бесплатный период или образовательный доступ.
- Развернута виртуальная машина (Compute Cloud) с настроенным SSH.
- Установлен и запущен Jupyter Notebook/Lab.
- Создан кластер Yandex Data Proc (Hadoop/Spark).
- Настроено подключение Google Colab к Yandex Cloud.

Критерии проверки:

- ✓ Можно подключиться к VM по SSH.
- ✓ Jupyter Notebook доступен через браузер.
- ✓ Spark-кластер активен, Zeppelin/Jupyter подключен.
- ✓ Данные из Object Storage загружаются в Colab.

Задача №2: Обработка данных

Действия:

- Загружены данные в RDD, выполнены операции `map`, `filter`, `reduce`.
- Реализован WordCount на RDD.
- CSV/JSON загружен в DataFrame, выполнены агрегации (`groupBy`, `agg`).
- Создана база в Managed MongoDB, выполнены CRUD-операции.
- Настроен Kafka-топик, Spark Streaming читает данные.

Критерии проверки:

- ✓ RDD-операции возвращают ожидаемый результат (например, WordCount корректно считает слова).

- ✓ DataFrame выводит корректные результаты после агрегации.
- ✓ В MongoDB есть записи, индексы ускоряют поиск.
- ✓ Spark Streaming выводит поток данных в реальном времени.

Задача №3: Аналитика и ML

Действия:

- Датасет (например, Iris) загружен в Spark.
- Данные разделены на train/test, обучена модель (логистическая регрессия/дерево решений).
- Оценена точность модели (accuracy, F1-score).
- Временной ряд загружен, построена ARIMA-модель.
- Сделаны прогнозы на будущие периоды.

Критерии проверки:

- ✓ Точность модели > 80% (для Iris).
- ✓ Прогноз ARIMA визуально соответствует тренду.
- ✓ Метрики (MAE, RMSE) рассчитаны.

Задача №4: Визуализация и интеграция

Действия:

- Построен интерактивный дашборд (Dash/Plotly).
- Добавлены фильтры (Dropdown, Slider).
- Дашборд развернут в облаке (например, Yandex Cloud Functions).
- Получены данные через Yandex Metrika API.
- Обработано аудио через SpeechKit (текст распознан).

Критерии проверки:

- ✓ Дашборд отображает данные без ошибок.
- ✓ Фильтры изменяют графики.
- ✓ Данные Metrika загружены в DataFrame.
- ✓ SpeechKit возвращает текст из аудио.

Задача №5: Оптимизация и масштабирование

Действия:

- Применено cache() / persist() к DataFrame.
- Данные разделены на партии.
- Замерено время выполнения запросов до/после оптимизации.
- Задание запущено в Yandex Data Proc через CLI.
- Ресурсы кластера (CPU, RAM) мониторятся.

Критерии проверки:

- ✓ Время выполнения запроса сократилось на 30-50%.
- ✓ Spark UI показывает закэшированные данные.
- ✓ CLI-команда успешно запускает задание.
- ✓ Нагрузка на кластер отображается в мониторинге.

Итоговый контроль

Для защиты работы студент предоставляет:

1. Скриншоты:

- Рабочего Jupyter/Colab с кодом.
- Результатов выполнения (графики, таблицы).
- Мониторинга кластера (CPU/RAM).

2. Файлы:

- Ноутбуки с кодом (.ipynb).

- Логи выполнения (если есть ошибки).

3. Устный ответ:

- Объяснение ключевых шагов.
- Анализ проблем и их решений.

Критерии оценки:

- Все пункты чек-листа выполнены.
- Результаты воспроизводимы.
- Студент может объяснить каждый этап.
- Защита проекта включает обоснование гипотез и интерпретацию результатов.
- Оценка за оформление финального отчета (20% от итоговой оценки).

Приведенный чек-лист предполагает гарантию полноценного освоения студентами лабораторных работ и подготовленности их к решению реальных задач в сфере Big Data.

Соответствие лабораторных работ и индикаторов компетенций (из РПД)

№	Тема	Задание	Соответствие индикаторам компетенций
1	Настройка окружения (Colab + Yandex Cloud)	Развернуть виртуальную машину в Yandex Cloud, подключиться через Jupyter.	ОПК-2.1 – применение системного подхода к анализу требований к инфраструктуре; BD-4.1 – осуществляет выбор технологий обработки больших данных, приемлемых для создания прикладной системы ИИ с заданными требованиями; BD-5.1 – осуществляет выбор направления вспомогательных технологических решений для формирования единого стека работы с большими данными
2	Основы PySpark	Загрузка данных в RDD, простые операции (map, filter, reduce).	BD-3.2 – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ с применением различных технологий хранения неструктурированных данных, оценивает качество; BD-4.1 – осуществляет выбор технологий обработки больших данных
3	Работа с Spark DataFrame	Анализ CSV-файла (агрегации, фильтрация).	BD-3.2 – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ; BD-4.1 – осуществляет выбор технологий обработки больших данных
4	Хранение данных в Yandex Object Storage	Загрузка и выгрузка данных из S3-совместимого хранилища.	BD-5.1 – осуществляет выбор направления вспомогательных технологических решений для формирования единого стека работы с большими данными
5	Парсинг и обработка логов	Анализ веб-логов с помощью Spark.	BD-3.2 – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ; LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость своих решений с точки зрения современного ИИ;
6	Работа с NoSQL (Yandex Managed MongoDB)	CRUD-операции, индексы.	BD-5.1 – осуществляет выбор направления вспомогательных технологических решений

7	Потоковая обработка (Kafka + Spark Streaming)	Чтение данных из Kafka-топика.	BD-4.1 – осуществляет выбор технологий обработки больших данных; PL-1.3 – разрабатывает и поддерживает системы обработки больших данных различной степени сложности
8	Машинное обучение в Spark (MLlib)	Классификация данных.	BD-4.1 – осуществляет выбор технологий обработки больших данных; LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость решений;
9	Визуализация данных (Dash/Plotly)	Построение дашборда.	LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость решений;
10	Оптимизация запросов в Spark	Кэширование, партиционирование.	BD-3.2 – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ с применением различных технологий хранения неструктурированных данных
11	Анализ социального графа (GraphX)	Поиск связей между узлами.	BD-4.1 – осуществляет выбор технологий обработки больших данных; LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость решений
12	Развертывание кластера (Yandex Data Proc)	Запуск Spark-задания в облаке.	BD-4.1 – осуществляет выбор технологий обработки больших данных; BD-5.1 – осуществляет выбор направления вспомогательных технологических решений
13	Анализ временных рядов	Прогнозирование с помощью ARIMA.	BD-4.1 – осуществляет выбор технологий обработки больших данных; LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость;
14	Облачная аналитика (Yandex Metrica API)	Выгрузка и анализ данных.	BD-4.1 – осуществляет выбор технологий обработки больших данных; BD-5.1 – осуществляет выбор направления вспомогательных технологических решений
15	Интеграция с API (Yandex SpeechKit)	Обработка аудио через облачный API.	BD-4.1 – осуществляет выбор технологий обработки больших данных
16	Финальный проект	Полный анализ датасета (от ETL до визуализации).	ОПК-2.1 – системный подход к анализу предметной области и выявлению требований к ИС; BD-3.2 – разрабатывает, отлаживает и тестирует прикладные решения с элементами ИИ; BD-4.1 – осуществляет выбор технологий обработки больших данных; LC-8.2 – определяет тенденции развития, оценивает новизну и практическую значимость;

Вывод

Соответствие лабораторных работ индикаторам компетенций подтверждает, что курс покрывает все ключевые аспекты Технологии обработки Big Data и ИИ: от написания запросов до развертывания кластеров и работы с алгоритмами машинного обучения.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

5.1 Основная литература:

1. Том Уайт – «Hadoop. Подробное руководство»
2. Холден Карау – «Изучаем Spark»
3. Мартин Клеппман – «Высоконагруженные приложения»
4. Алексей Губанов – «NoSQL. Нереляционные базы данных»
5. Андрей Макаров – «Big Data и машинное обучение»
6. Алексей Натекин. Большие данные: принципы обработки.
7. Гудков А.В. – "Обработка больших данных".
8. Райков А.Н. – "Apache Spark для аналитики данных".
9. Камкин А.С. – "NoSQL: принципы и практика".

5.2 Дополнительная литература:

1. Официальная документация Apache Spark, Kafka.
2. Yandex Cloud – руководства по Data Proc.
3. Харрисон Д. – "Python и анализ данных".
4. Колдаев, В. Д. Структуры и алгоритмы обработки данных : учебное пособие / В. Д. Колдаев. - Москва : РИОР : ИНФРА-М, 2021. - 296 с. - ЭБС ZNANIUM.com. - URL: <https://znanium.com/catalog/product/1230215> (дата обращения: 07.12.2023). – Текст : электронный.
5. Сенько, А. Работа с BIGDATA в облаках. Обработка и хранение данных с примерами из Microsoft / А. Сенько. - СПб.: Питер, 2019. - 448 с. - ЭБС ZNANIUM.com. - URL: <https://znanium.com/catalog/product/1230215> (дата обращения: 07.12.2023). – Текст : электронный.
6. Нархид, Н. Apache Kafka. Поточковая обработка и анализ данных / Н. Нархид. - СПб.: Питер, 2019. - 320 с. ЭБС ZNANIUM.com. - URL: <https://znanium.com/catalog/product/1230215> (дата обращения: 07.12.2023). – Текст: электронный.
7. Сенько, А. Работа с BigData в облаках. Обработка и хранение данных с примерами из Microsoft Azure / А. Сенько. - СПб.: Питер, 2019. - 448 с. ЭБС Университетская библиотека ONLINE. – URL: <https://biblioclub.ru/index.php?page=book&id=598404> (дата обращения: 07.12.2023). – Текст : электронный.
8. Sun, X., Li, J., Kovalenko, A.V., Feng, W., Ou, Y. Integrating Reinforcement Learning and Learning From Demonstrations to Learn Nonprehensile Manipulation //IEEE Transactions on Automation Science and Engineering, 2023, 20(3), 1735–1744, DOI: 10.1109/TASE.2022.3185071, Q1
9. Petukhova, A.V.; Kovalenko, A.V.; Ovsyannikova, A.V. Algorithm for Optimization of Inverse Problem Modeling in Fuzzy Cognitive Maps. Mathematics 2022, 10, 3452. DOI: 10.3390/math10193452, Q1
10. Kirillova, E.; Kovalenko, A.; Urtenov, M. Study of the Current–Voltage Characteristics of Membrane Systems Using Neural Networks. AppliedMath 2025, 5, 10. <https://doi.org/10.3390/appliedmath5010010>
11. Kadurin, Artur, et al. "The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology." Oncotarget 8.7 (2016): 10883.
12. Kadurin, Artur, et al. "druGAN: an advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico." Molecular pharmaceuticals 14.9 (2017): 3098-3104.
13. Polykovskiy, Daniil, et al. "Molecular sets (MOSES): a benchmarking platform for molecular generation models." Frontiers in pharmacology 11 (2020): 565644.
14. Khrabrov, Kuzma, et al. " ∇^2 DFT: A Universal Quantum Chemistry Dataset of Drug-Like Molecules and a Benchmark for Neural Network Potentials." Advances in Neural Information Processing Systems 37 (2024): 36869-36889.
15. Polykovskiy, Daniil, et al. "Entangled conditional adversarial autoencoder for de novo drug discovery." Molecular pharmaceuticals 15.10 (2018): 4398-4405.
16. Николенко, Сергей, Кадури, Артур и Архангельская Екатерина. Глубокое обучение. Издательский дом "Питер", 2017.

5.3. Периодические издания и конференции (А*):

1. IEEE Transactions on Big Data – научные статьи по обработке больших данных.
2. Journal of Big Data (SpringerOpen) – открытый журнал с исследованиями в области Big Data.
3. Big Data Research (Elsevier) – публикации по анализу, управлению и визуализации данных.
4. Data Science Journal (CODATA) – междисциплинарные исследования данных.
5. ACM Transactions on Knowledge Discovery from Data (TKDD) – методы извлечения знаний из больших данных.
6. <https://openreview.net/forum?id=FMMF1a9ifL>
7. <https://openreview.net/forum?id=ElUrNM9U8c#discussion>
8. <https://openreview.net/forum?id=JoO6mtCLHD>
9. <https://aclanthology.org/2024.findings-emnlp.760/>
10. <https://aclanthology.org/2020.coling-main.588/>
11. https://link.springer.com/chapter/10.1007/978-3-030-72113-8_30
12. https://link.springer.com/chapter/10.1007/978-3-031-42448-9_10
13. <https://aclanthology.org/2024.findings-naacl.288/>

5.4. Интернет-ресурсы, в том числе современные профессиональные базы данных и информационные справочные системы

Базы данных и аналитические платформы

1. Google BigQuery – облачная аналитика больших данных.
2. Apache Hadoop & Spark – официальная документация и ресурсы.
3. Kaggle – датасеты, соревнования и учебные материалы.
4. Cloudera – платформа для работы с Big Data.
5. Databricks – решения на основе Apache Spark.

Справочные системы и блоги

1. Towards Data Science (Medium) – статьи по Data Science и Big Data.
2. KDnuggets – новости, обучающие материалы и обзоры инструментов.
3. O'Reilly Data & AI – книги и статьи по Big Data и машинному обучению.
4. IBM Big Data Hub – кейсы и руководства по Big Data.

Ресурсы свободного доступа

1. [Apache Spark Documentation](#)
2. [Yandex Cloud Big Data](#)
3. [Kaggle Datasets](#)

Собственные электронные образовательные и информационные ресурсы КубГУ

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

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

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная учебная работа (консультации) – дополнительное разъяснение учебного материала.

Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или лицом с ограниченными возможностями здоровья.

По курсу предусмотрено проведение лекционных занятий, на которых дается систематизированный материал по технологиям обработки больших данных. В ходе лекций рассматриваются ключевые концепции.

Лабораторные занятия курса посвящены практическому освоению технологиям обработки больших данных

При самостоятельной работе студентам необходимо изучать рекомендованную литературу в виде официальной документации к используемым открытым программным продуктам, облачным платформам.

Важнейшим компонентом курса является самостоятельная проектная работа, в ходе которой студент разрабатывает законченное решение для решения задач (кейсов) промышленных партнеров. Допускается выполнение проектов в командах.

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

Ключевые аспекты взаимодействия с промышленными партнерами:

- Для УГТ 5 – ИП помогает определить реалистичные условия тестирования, но не рискует своей инфраструктурой.
- Для УГТ 6 – ИП предоставляет "песочницу" или изолированную среду, где можно выявить скрытые проблемы.
- Для УГТ 7 – ИП становится соразработчиком, так как технология адаптируется под его конкретные процессы.

А. Применение технологий обработки больших данных в кейсах ПАО «Сбербанк»

1. Прогнозирование оттока клиентов (Churn Prediction)

Описание:

Анализ поведения клиентов для выявления тех, кто с высокой вероятностью может уйти к конкурентам.

Цель: Снизить отток клиентов на 15-20% за счет персональных предложений.

Технологии:

✓ **Spark MLlib** (CatBoost, XGBoost)

✓ **Hadoop HDFS** (хранение истории транзакций)

✓ **Tableau** (визуализация результатов)

Реализация:

а) Сбор данных: история транзакций, активность в приложении, обращения в поддержку.

b) Обучение модели на признаках:

- ✓ Снижение активности
- ✓ Уменьшение количества операций
- ✓ Жалобы в чате поддержки (NLP-анализ)

c) Интеграция с CRM-системой для автоматических предложений.

Результат:

- ✓ Точность модели: **87%**
- ✓ Снижение оттока: **18%** за 6 месяцев
- ✓ Автоматизированные триггеры для маркетинга (например, cashback для "группы риска").

2. Real-time Anti-Fraud для платежей

Описание: Система для мгновенного выявления мошеннических операций.

Цель: Снизить ущерб от мошенничества на 30%.

Технологии:

- ✓ **Apache Kafka** (поток транзакций)
- ✓ **Spark Streaming** (анализ в реальном времени)
- ✓ **GraphX** (поиск связей между счетами)

Реализация:

a) Настройка Kafka-топика для транзакций (100K+ событий/сек).

b) Алгоритмы обнаружения аномалий:

- ✓ Необычные суммы/места операций
- ✓ Повторяющиеся переводы на новые счета
- c) Автоматическая блокировка подозрительных операций.

Результат:

- ✓ Скорость обработки: **<100 мс** на операцию
- ✓ Снижение фрода: **35%**
- ✓ Интеграция с ЦБ РФ для отчетности.

3. Персонализация предложений в мобильном приложении

Описание: ИИ-система для рекомендации финансовых продуктов.

Цель: Увеличить конверсию в продажах на 25%.

Технологии:

- ✓ **Apache Spark** (анализ поведения)
- ✓ **Redis** (кеширование рекомендаций)
- ✓ **A/B-тестирование** (оптимизация алгоритмов)

Реализация:

a) Сбор данных:

- ✓ История покупок
- ✓ Геолокация

✓ Время активности

b) Коллаборативная фильтрация + CatBoost.

c) Динамический интерфейс в приложении.

Результат:

- ✓ Рост продаж кредитных карт: **28%**
- ✓ Увеличение среднего чека: **15%**

4. Оптимизация работы колл-центра с NLP

Описание: Автоматизация обработки обращений клиентов.

Цель: Сократить нагрузку на операторов на 40%.

Технологии:

- ✓ **BERT/GPT-3** (классификация запросов)
- ✓ **Kafka** (поток аудио/текста)
- ✓ **Yandex SpeechKit** (STT/TTS)

Реализация:

- a) Т р а н с к р и п ц и я з в о н к о в → т е к с т.
- b) Классификация интенгов (жалобы, вопросы по картам и т.д.).
- c) Автоответы через чат-бота.

Результат:

- ✓ **60%** обращений решается без оператора
- ✓ Снижение времени ответа: с 5 мин до **30 сек**

5. Оптимизация сети банкоматов с геоаналитикой

Описание: Анализ расположения и загрузки банкоматов.

Цель: Сократить затраты на инкассацию на 20%.

Технологии:

- ✓ **GeoSpark** (обработка геоданных)
- ✓ **H3 Uber Hexagons** (кластеризация)
- ✓ **Kepler.gl** (визуализация)

Реализация:

- a) Сбор данных:
 - ✓ Транзакции по координатам
 - ✓ График инкассации
- b) Поиск "мертвых" банкоматов.
- c) Оптимизация маршрутов.

Результат:

- ✓ Сокращение банкоматов: **12%**
- ✓ Экономия: **200 млн руб./год**

Итоговая таблица эффективности

Кейс	Технологии	Экономический эффект
Прогнозирование оттока	Spark ML, Hadoop	+18% удержание
Anti-Fraud	Kafka, GraphX	-35% фрод
Персонализация	Spark, Redis	+28% продажи
NLP-колл-центр	BERT, SpeechKit	-60% нагрузка
Оптимизация АТМ	GeoSpark, H3	200 млн руб./год

Вывод: Сбербанк использует «Технологии обработки больших данных» для:

- ✓ **Риск-менеджмента** (фрод, скоринг)
- ✓ **Маркетинга** (персонализация)

- ✓ **Оптимизации** (логистика, автоматизация)

Лабораторные работы можно адаптировать под эти кейсы, используя **PySpark, Kafka и ML-библиотеки**.

Б. Применение технологий обработки больших данных в кейсах компании AVA LAB

1. Обнаружение мошеннических транзакций в реальном времени

Описание: Разработка системы для выявления подозрительных платежей в фин-тех-приложениях.

Цель: Снизить ущерб от мошенничества на 40% с задержкой обработки <100 мс.

Технологии:

- ✓ **Apache Kafka** (потокковая передача транзакций)
- ✓ **Spark Structured Streaming** (анализ в реальном времени)
- ✓ **GraphX** (выявление связанных аккаунтов)
- ✓ **CatBoost** (ML-модель для аномалий)

Реализация:

a) Настройка Kafka-топика для приема транзакций (до 50К событий/сек).
b) Обучение модели на исторических данных с метками "мошенничество/легитимно".

c) Развертывание Spark-джобы для потоковой обработки.
d) Интеграция с графовой БД (Neo4j) для визуализации связей.

Результат:

- ✓ Точность детекции: **92%**
- ✓ Снижение фрода: **45%**
- ✓ Скорость обработки: **80 мс**

2. Оптимизация кредитного скоринга для МФО

Описание: Скоринговая система на основе альтернативных данных (цифровой след, соцсети).

Цель: Увеличить одобрение кредитов надежным заемщикам на 25%.

Технологии:

- ✓ **PySpark ML** (Feature Engineering)
- ✓ **HDFS** (хранение сырых данных)
- ✓ **SHAP** (интерпретируемость модели)

Реализация:

a) Сбор данных: история браузинга, геолокация, активность в соцсетях (с согласия).

b) Обучение Gradient Boosting-модели с учетом регуляризации.

c) Разработка дашборда в **Superset** для анализа решений.

Результат:

- ✓ Увеличение approval rate: **+28%**
- ✓ Снижение дефолтов: **15%**
- ✓ Автоматизированное принятие решений для 80% заявок.

3. NLP-анализ голосовых обращений в колл-центр

Описание: Автоматизация обработки жалоб клиентов через speech-to-text и классификацию интенгов.

Цель: Сократить затраты на колл-центр на 35%.

Технологии:

- ✓ **Yandex SpeechKit** (расшифровка аудио)
- ✓ **BERT** (классификация текста)
- ✓ **Airflow** (оркестрация pipeline)

Реализация:

- а) Транскрипция звонков в текст (русский язык + диалекты).
- б) Обучение BERT-модели на размеченных данных (15 категорий: "жалоба", "запрос информации" и т.д.).
- с) Интеграция с CRM для автоматических ответов.

Результат:

- ✓ Точность классификации: **89%**
- ✓ Сокращение ручной обработки: **60%**
- ✓ Среднее время ответа: **20 сек** (было 5 мин).

4. AML-аналитика для криптобирж

Описание: Выявление схем отмывания денег через анализ цепочек транзакций в блокчейне.

Цель: Обнаруживать 95% подозрительных операций.

Технологии:

- ✓ **Spark GraphFrames** (анализ графа транзакций)
- ✓ **Temporal Graph Networks** (учет временных меток)
- ✓ **Elasticsearch** (быстрый поиск паттернов)

Реализация:

- а) Парсинг blockchain-данных (Bitcoin/Ethereum) в графовую структуру.
- б) Поиск циклических переводов и "мусорных" кошельков.
- с) Визуализация схем в **Gephi**.

Результат:

- ✓ Обнаружено **1200+** подозрительных кластеров
- ✓ Интеграция с регуляторами (ЦБ, FATF)

5. Персонализация fintech-приложений

Описание: Рекомендательная система для финансовых продуктов на основе поведения пользователей.

Цель: Увеличить конверсию в покупку продуктов на 30%.

Технологии:

- ✓ **Apache Flink** (обработка событий в реальном времени)
- ✓ **Redis** (кеширование рекомендаций)
- ✓ **Bandit-алгоритмы** (A/B-тестирование)

Реализация:

- а) Сбор данных: клики, время в приложении, демография.
- б) Обучение hybrid-модели (коллаборативная фильтрация + content-based).
- с) Динамическое обновление рекомендаций каждые **5 мин**.

Результат:

- ✓ Рост продаж: **32%**
- ✓ Увеличение среднего чека: **18%**

Сводная таблица результатов

Кейс	Ключевые технологии	Эффективность
------	---------------------	---------------

Anti-Fraud	Kafka, Spark, GraphX	-45% фрод, 92% точность
Кредитный скоринг	PySpark, SHAP	+28% одобрений
NLP-колл-центр	BERT, SpeechKit	-60% затрат, 89% accuracy
AML для крипто	GraphFrames, Gephi	1200+ схем обнаружено
Персонализация	Flink, Redis	+32% конверсия

Вывод: AVA LAB использует «Технологии обработки больших данных» для:

- ✓ **Безопасности** (Anti-Fraud, AML)
- ✓ **Финансовой аналитики** (скоринг, рекомендации)
- ✓ **Автоматизации** (NLP, потоковая обработка)

Для лабораторных работ:

1. Реализовать детектор аномалий на синтетических транзакциях.
2. Построить граф связей для AML-анализа.
3. Обучить BERT-модель для классификации текстов.

Инструменты: **Spark, Kafka, Python (PySpark), JupyterHub.**

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

7.1 Перечень информационно-коммуникационных технологий

- Облачные платформы (Google Cloud, AWS, Microsoft Azure, Yandex Cloud)
- Распределённые системы хранения и обработки данных (HDFS, S3, HBase, Cassandra)
- Технологии потоковой обработки данных (Apache Kafka, Apache Flink, Apache Storm)

Системы управления базами данных (СУБД)

- Реляционные (PostgreSQL, MySQL)
- NoSQL (MongoDB, Redis, Elasticsearch)
- Фреймворки для распределённых вычислений (Apache Hadoop, Apache Spark)
- Инструменты визуализации данных (Tableau, Power BI, Apache Superset, Grafana)
- Контейнеризация и оркестрация (Docker, Kubernetes)
- Средства мониторинга и управления инфраструктурой (Prometheus, Grafana, ELK Stack)
- API и веб-сервисы (REST, GraphQL, gRPC)

7.2 Перечень лицензионного и свободно распространяемого программного обеспечения

Лицензионное ПО:

Интегрированные среды разработки (IDE):

JetBrains IntelliJ IDEA (с поддержкой Scala, Python, Java)
PyCharm Professional (для Python-разработки)
Microsoft Visual Studio (с инструментами для Big Data)

Корпоративные решения для Big Data:

Cloudera Data Platform (CDP)
Microsoft SQL Server (с поддержкой Big Data)

Облачные сервисы (платные подписки):

- Google BigQuery

- AWS EMR (Elastic MapReduce)
- Azure HDInsight

Свободно распространяемое (open-source) ПО:

Обработка и анализ данных:

Apache Hadoop (HDFS, MapReduce, YARN)
Apache Spark (для быстрой обработки данных)
Apache Flink (поточковая обработка)
Apache Kafka (распределённый потоковый брокер)

Базы данных и хранилища:

PostgreSQL (+ расширение TimescaleDB для временных рядов)
MongoDB (документноориентированная NoSQL)
Apache Cassandra (высокомасштабируемая NoSQL)
Redis (ключ-значение, кэширование)
Elasticsearch (поиск и аналитика)

Визуализация и BI-инструменты:

Apache Superset (альтернатива Tableau)
Grafana (мониторинг и дашборды)
Metabase (open-source BI)

Разработка и управление инфраструктурой:

Jupyter Notebook / JupyterLab (интерактивная аналитика)
Docker (контейнеризация)
Kubernetes (оркестрация контейнеров)
Apache Airflow (оркестрация ETL-процессов)

Языки программирования и библиотеки:

- **Python** (Pandas, NumPy, SciPy, Scikit-learn, PySpark)
- **R** (для статистического анализа)
- **Scala** (работа с Apache Spark)
- **SQL** (для работы с базами данных)

Дополнительные инструменты

- **Git** (система контроля версий, GitHub/GitLab/Bitbucket)
- **Apache Zeppelin** (аналитика и визуализация в браузере)
- **MLflow** (управление машинным обучением)
- **Apache NiFi** (автоматизация потоков данных)

8. Материально-техническое обеспечение по дисциплине (модулю)

Виртуальные машины, кластер Managed Kubernetes и ресурсы GPU в облаке предоставляется индустриальным партнером ПАО «Сбербанк»:

№	Продукт	Параметры продукта	Кол-во	Кол-во кон-фигу-раций	Ед. изм.
1	Виртуальная машина	Виртуальная машина 10% vCPU 2 vCPU 4 RAM	1	60	Шт
		ОС Ubuntu 22.04	1		Шт
		Системный диск SSD	1		Шт

			10		Гб
		Аренда публичного IP	1		Шт
2	Виртуальная машина с GPU	Виртуальная машина с GPU NVIDIA® Tesla® V100 2 GPU 8 vCPU 128 Гб RAM	1	1	Шт
		ОС Ubuntu_24.04	1		Шт
		Системный диск SSD	1		Шт
			2000		Гб
		Диск SSD	1		Шт
			4096		Гб
		Диск SSD	1		Шт
			4096		Гб
		Аренда публичного IP	1		Шт
3	K8S	Master node 8 vCPU 16 RAM	1	1	Шт
		Worker node 10% доля 4 vCPU 32 RAM	5		Шт
		Worker node SSD-NVME	64		Гб
		Аренда публичного IP	1		Шт
4	ML Inference Instance Type GPU	Время работы в месяц	40	1	Ч
		Инстанс 8 x NVIDIA® H100 NVLink PCIe 160 vCPU 1520 GB RAM	1		Шт
		Количество запросов к ML-моделям	1		Млн. Шт
		Кэш ML-моделей	160		Гб
5	LLM	Токены GigaChat 2 Max	50		Млн. Шт
		Токены Embeddings	400		Млн. Шт

Дополнительные облачные ресурсы предоставляются технологическим партнером Yandex Cloud.

№	Вид работ	Наименование учебной аудитории, ее оснащенность оборудованием и техническими средствами обучения
1.	Лекционные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
2.	Лабораторные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, проектором, программным обеспечением
3.	Практические занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
4.	Групповые (индивидуальные) консультации	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
5.	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением

6.	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.
----	------------------------	--

Примечание: Конкретизация аудиторий и их оснащение определяется ОПОП.