ОРКЕСТРАЦИЯ МОДЕЛЕЙ МАШИННОГО ОБУЧЕНИЯ ДЛЯ РЕШЕНИЯ ЗАДАЧ АНАЛИЗА ПРОСТРАНСТВЕННЫХ ДАННЫХ[1]

Куманькин Д.С., Ямашкин С.А.
ФГБОУ ВО «МГУ им. Н.П. Огарёва», Саранск, Россия
d.kumankin@gmail.com

Аннотация. В данной статье представлены результаты исследований, посвящённых архитектурам и компонентам систем оркестрации моделей машинного обучения, направленных на решение задач анализа пространственных данных. Рассмотрены этапы жизненного цикла моделей, определены ключевые компоненты систем оркестрации, а также их функционал. Предложена архитектура оркестратора, включающая рассмотренные компоненты и схему их взаимодействия. В ходе исследования разработано более пятидесяти различных сценариев, касающихся процессов работы разрабатываемой системы оркестрации.

Ключевые слова: оркестрация моделей машинного обучения, MLOps, архитектура систем машинного обучения, конвейеры машинного обучения, пространственные данные.

Введение

В настоящее время системы машинного обучения (ML) активно используются для решения различных задач по анализу и обработке геопространственных данных. При этом происходит активный рост сложности таких систем, что в свою очередь требует детальной проработки их архитектур и сценариев использования. В данной статье представлены результаты научно-исследовательской работы задачей которой являлся анализ архитектурных подходов, направленных на проектирование и разработку систем оркестрации моделей машинного обучения, для решения задач анализа пространственных данных.

Системы оркестрации моделей машинного обучения позволяют управлять процессами конвейера машинного обучения и автоматизировать выполнение задач их жизненного цикла. Они осуществляют развёртывание моделей, управляют ресурсами и позволяют отслеживать поведение моделей. Такие системы должны быть легко масштабируемы, надёжны, а их компоненты быть совместимы.

Стоит обратить внимание на то, что ядро машинного обучения является лишь малой частью таких систем. Комплексные системы машинного обучения для анализа геопространственных данных содержат множество компонентов, которые обеспечивают этапы загрузки данных, их предварительной обработки, хранения и множество других сценариев. Они позволяют полностью управлять конвейером обучения модели и отслеживать поведение системы.

1. Жизненный цикл моделей машинного обучения

Для понимания того, как устроены современные системы машинного обучения и процессы их оркестрации необходимо рассмотреть этапы жизненного цикла моделей машинного обучения. На рисунке 1 представлен жизненный цикл моделей машинного обучения, который состоит из нескольких последовательных этапов, каждый из которых играет важную роль в создании качественной и эффективной модели [1].

На первом этапе жизненного цикла моделей машинного обучения (выделено цветом на рисунке 1) проводится загрузка, версионирование данных и их хранение. Затем полученные данные подвергаются процессам проверки и очистки, это необходимо для того, чтобы избежать ошибок в ходе обучения моделей, так как без качественных данных ни один алгоритм машинного обучения не может гарантировать надлежащий результат. Следующим этапом является предварительная обработка данных, в ходе которой данные приводятся к требуемому для выбранной модели машинного обучения виду [2]. После этого модель обучается на подготовленных данных с использованием выбранных алгоритмов машинного обучения. Далее проводится анализ и оценка модели, где она подвергается оценкам по различным метрикам качества и производительности. Проверка модели осуществляется на тестовой выборке данных для оценки её обобщающей способности, что помогает убедиться в её надлежащей работе на новых, незнакомых ей ранее данных [1, 2].

Обучение моделей происходит циклично, в несколько итераций, в ходе которых производится настройка параметров модели, которая подразумевает оптимизацию гиперпараметров для улучшения

[1] *Исследование выполнено в рамках научно-исследовательского проекта «Создание лаборатории искусственного интеллекта» программы социально-экономического развития Республики Мордовия на 2022-2026 годы.*

производительности модели [2]. По окончанию такого цикла выбирается модель, которая наиболее удовлетворяет заданным требованиям. После этого модель может быть использована для работы с реальными данными.

Заключительным этапом, который часто упускают из виду, является система обратной связи. Она должна собирать и обрабатывать данные о работе модели в реальных условиях и отзывы пользователей. На основе этой информации могут быть внесены изменения и улучшения, что в свою очередь позволяет совершенствовать модели [1]. Все рассмотренные выше этапы обеспечивают комплексный подход к созданию и эксплуатации моделей машинного обучения.

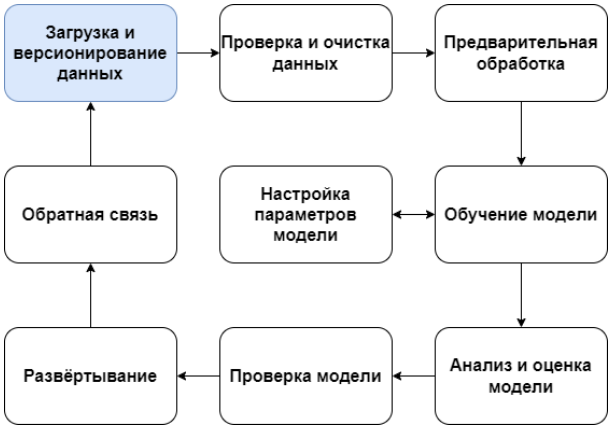

Рис. 1. Жизненный цикл модели машинного обучения [1]

2. Компоненты систем оркестрации

Существует большое число различных систем оркестрации, таких как Kubernetes, Apache Airflow, MLflow и TensorFlow Extended. Каждая из этих систем имеет свои особенности, функции и подходы к управлению рабочими процессами. Эти системы предоставляют множество инструментов для интеграции, планирования, выполнения и мониторинга задач, что существенно повышает эффективность и надежность процессов машинного обучения. На основе проведённого исследования различных систем оркестрации были определены основные компоненты таких систем и их предназначение (рисунок 2).

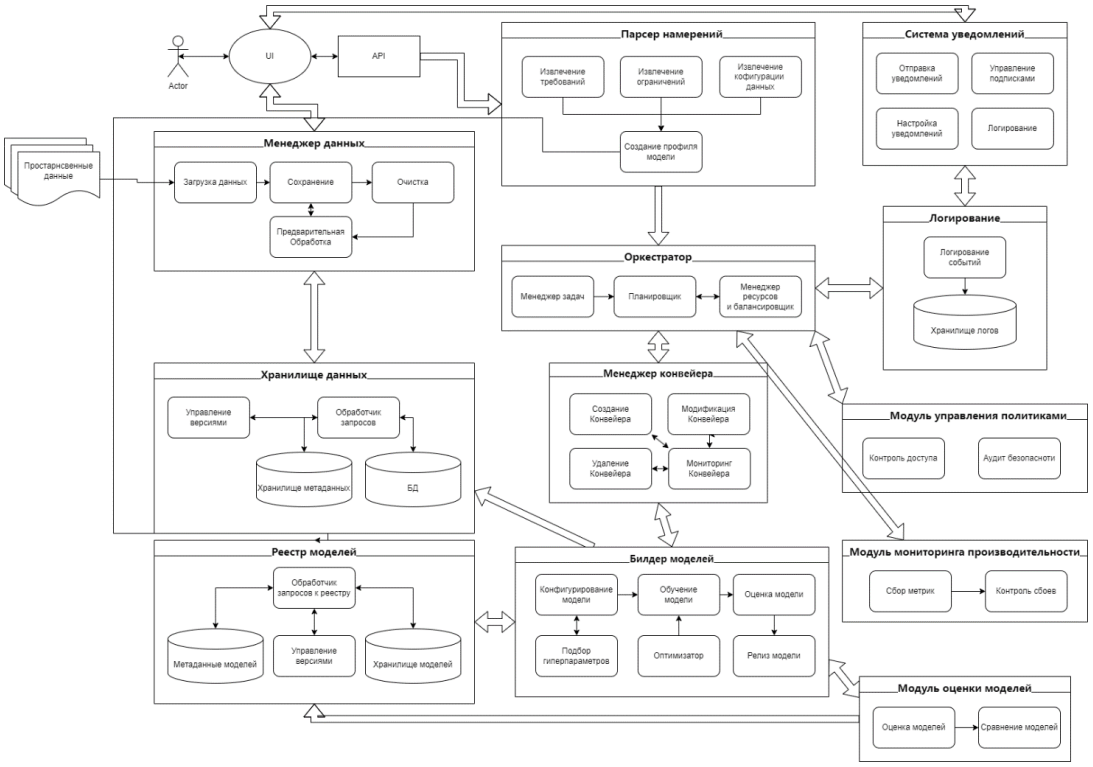

Рис. 2. Диаграмма компонентов системы оркестрации

Оркестратор является центральным компонентом, и обеспечивает выполнение различных этапов процесса машинного обучения. Он обеспечивает управление другими компонентами системы, занимается планированием и запуском задач, а также контролем ресурсов. Основными подкомпонентами оркестратора являются: планировщик, менеджер задач и менеджер ресурсов. Планировщик определяет порядок выполнения задач, распределяет задачи между доступными ресурсами и обеспечивает их последовательное и параллельное выполнение. Менеджер задач управляет задачами, отслеживает их статус и прогресс, а в случае сбоя перезапускает выполнение задачи. Менеджер ресурсов должен обеспечивать контроль и управление вычислительными ресурсами системы (CPU, GPU, RAM и т.п.), оптимизировать распределение ресурсов между задачами конвейера, и по мере необходимости производить скалирование.

Хранилище данных обеспечивает хранение и доступ к геопространственным данным. Оно сохраняет как «сырые» (необработанные), так и обработанные данные, предоставляет доступ к данным на основе политик доступа, а также обеспечивает присвоение версий сохраняемым данным [3].

Менеджер данных должен позволять проводить загрузку данных из подключаемых источников, на основе архитектурного потерна Адаптер (Adapter). Загруженные данные сначала должны быть проверены на целостность и корректность, а затем сохранены в хранилище данных с присвоением им версии. Далее, по запросу оркестратора, менеджер данных проводит очистку данных и их предварительную обработку под конкретную модель.

Реестр моделей предназначен для хранения моделей машинного обучения и обеспечивает доступ к ним на основе политик доступа. Помимо моделей в нём хранятся метаданные моделей, такие как параметры обучения, указатели к входным данным, результаты мониторинга производительности и оценки моделей [3].

Билдер моделей реализует песочницу для циклического обучения моделей, на основе полученной от оркестратора задачи, которая содержит инструкции для работы конвейера машинного обучения. В песочнице реализуется выбор модели, загрузка предварительно подготовленных данных из хранилища данных, настройка гиперпараметров и обучение модели. Обученная модель подвергается оценке на основе метрик производительности и качества, и на основе этого принимается решение о возможности релиза модели с присвоением версии и последующим сохранением её в реестр.

Модуль оценки моделей позволяет оценивать качество и производительность обученной модели, сравнивать результаты и создавать отчёты об оценке моделей.

Модуль мониторинга производительности следит за производительностью моделей, обнаруживает аномалии и ухудшение качества моделей, и обеспечивает мониторинг использования ресурсов для повышения стабильности системы.

Модуль логирования и трассировки позволяет вести логи событий, для дальнейшей диагностики поведения системы. Он должен обеспечивать запись в лог файл всех событий, происходящих в системе, а также трассировку хода выполнения задач по обучению моделей.

Система уведомлений – отправляет уведомления о различных событиях. Она уведомляет пользователей о статусе задачи, отправляет предупреждения о проблемах и отказах, обеспечивает возможности настройки и фильтрации уведомлений.

Менеджер конвейера позволяет создавать, изменять, удалять и просматривать конвейеры машинного обучения. Управлять конвейерами обработки данных и обучения моделей, обеспечивая гибкость построения и изменения конвейеров на основе различных паттернов проектирования конвейеров.

На основе запросов пользователя парсер намерений преобразует высокоуровневые намерения пользователя в детальные задачи для оркестратора. Он формирует требования и ограничения для конвейера и моделей, определяет модель поведения для данных и подготавливает профиль модели машинного обучения.

Менеджер управления политиками – обеспечивает управление политиками доступа для обеспечения безопасности.

3. Сценарии процессов работы системы оркестрации

Проведенный анализ функционала компонентов систем оркестрации и их взаимодействия между собой, анализ требований к таким компонентам, а также понимание этапов жизненного цикла моделей машинного обучения позволили разработать более пятидесяти различных сценариев поведения системы оркестрации для решения задач анализа пространственных данных. Рассмотрим лишь некоторые из таких сценариев.

На рисунке 3 представлена диаграмма активности процесса загрузки данных космической съёмки. Она описывает последовательность действий, необходимых для успешного выполнения задачи загрузки данных. Начальным состоянием является ожидание запроса на загрузку данных. После получения запроса система переходит к этапу загрузки данных. На этом этапе данные загружаются из внешнего источника, такого как хранилище данных космических снимков. После загрузки данных система проверяет их целостность и корректность. В случае успешной валидации данные сохраняются в хранилище данных, например, в формате GeoTIFF.

Примером такого процесса может быть загрузка изображений Sentinel-2 с платформы Copernicus Open Access Hub. Пользователь отправляет запрос на загрузку данных с указанием параметров, таких как диапазон дат, координаты интересующей области и т.п. Далее система, используя API платформы ищет и загружает требуемые изображения. В случае успешной загрузки система сообщает об успехе, а в случае возникновения ошибок на любом этапе процесса система генерирует и отправляет уведомление об ошибке, детализируя причину сбоя.

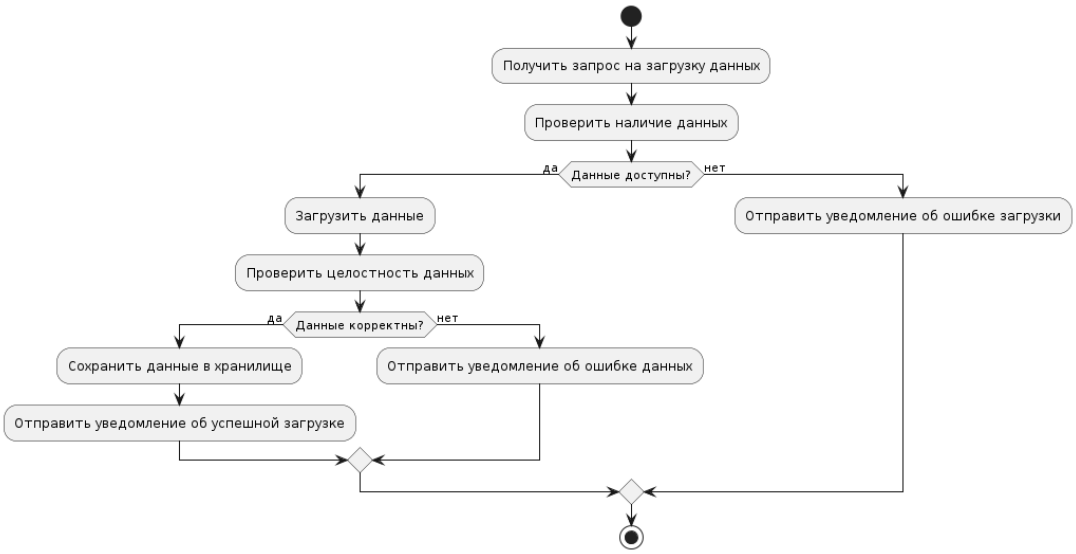

Рис. 3. Диаграмма активности процесса загрузки данных

Процесс обновления модели является важнейшим компонентом системы управления моделями машинного обучения, обеспечивающим актуальность моделей на основе новых данных и улучшенных алгоритмов. Схемы действий этого процесса представлены на рисунках 4 и 5, они описывают последовательность действий, необходимых для успешного выполнения задачи обновления модели. Начальным состоянием является ожидание запроса на обновление модели. После получения запроса система загружает текущую версию модели из реестра моделей. Затем она инициирует запрос на загрузку новых обучающих данных для модели. На этапе проверки данных система проверяет целостность и корректность новых данных. Если проверка данных прошла успешно, система переходит к обучению обновляемой модели [4].

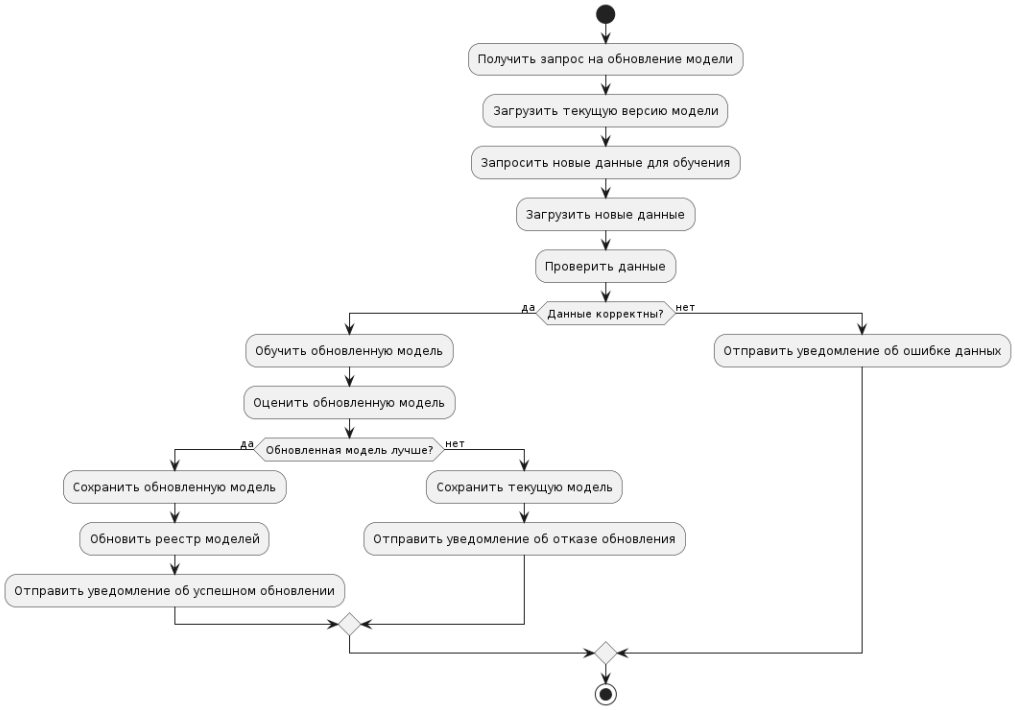

Рис. 4. Диаграмма активности процесса обновления модели

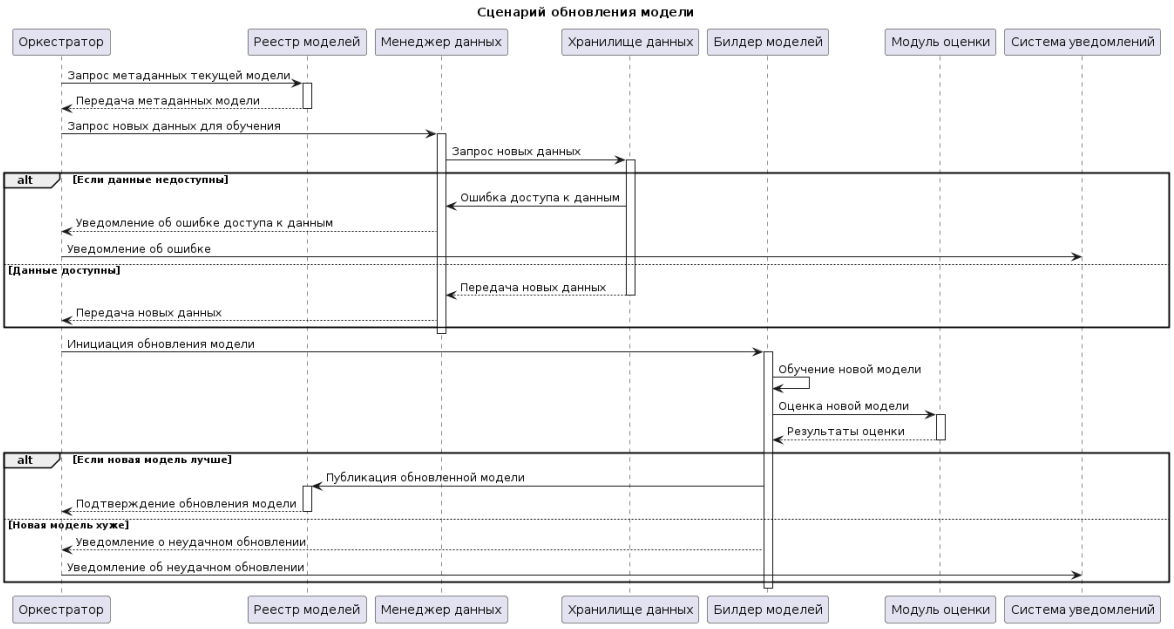

Рис. 5. Диаграмма последовательности процесса обновления модели

Процесс автоматического тестирования моделей является ключевым элементом в системе управления моделями машинного обучения, обеспечивая эффективное управление задачами и их ресурсами [5]. Схема действий этого процесса представлена на рисунке 6, она описывает последовательность действий, необходимых для успешного планирования и выполнения задач. Исходное состояние – ожидание запроса на планирование задачи. После получения запроса система переходит к определению порядка выполнения задач.

На этом этапе планировщик определяет зависимости между задачами и их приоритеты. Далее задачи распределяются по доступным ресурсам, с учетом текущей нагрузки и требований к задаче. Затем задачи запускаются, и система переходит к мониторингу их выполнения.

Мониторинг выполнения задач состоит из этапов отслеживания статуса каждой задачи и сбора показателей производительности. При обнаружении ошибок система автоматически переходит к этапу обработки ошибок, где задачи могут быть перезапущены или применены другие корректирующие действия.

Рис. 6. Диаграмма активности процесса автоматического тестирования

Процесс обработки сбоев в процессе обучения модели является важным для обеспечения надежности и устойчивости системы оркестрации моделей машинного обучения. Диаграмма последовательности (рисунок 7) этого сценария описывает взаимодействие между различными компонентами системы для обработки ошибки, возникающей во время обучения модели.

Оркестратор получает запрос на обучение модели и передает его менеджеру данных для подготовки данных к обучению. Менеджер данных загружает и подготавливает данные, передавая их билдеру моделей. Билдер моделей начинает процесс обучения модели. В случае возникновения ошибки билдер моделей сообщает об ошибке оркестратору [7, 8]. Оркестратор фиксирует ошибку, логирует событие, отправляет уведомление через систему уведомлений и пытается перезапустить процесс обучения, либо применяет альтернативные меры согласно определённым сценариям.

Рис. 7. Диаграмма последовательности сценария обработки сбоя в процессе обучения модели

4. Заключение

В статье представлены результаты исследований архитектур и компонентов систем оркестрации машинного обучения для решения задач анализа пространственных данных. Рассмотрены этапы жизненного цикла моделей машинного обучения. Определены основные компоненты систем оркестрации и их функционал. Предложена архитектура оркестратора, содержащего рассмотренные компоненты и схема их взаимодействия. Разработано более пятидесяти различных сценариев касающихся процессов работы разрабатываемой системы оркестрации.

Применение систем оркестрации необходимо для реализации полного жизненного цикла моделей машинного обучения. Системы оркестрации позволяют управлять процессом обучения моделей, их мониторингом, развёртыванием и управлением вычислительными ресурсами.

Область применения исследований: системы машинного обучения для решения задач анализа пространственных данных.

Литература

1. *Куманькин Д.С., Ямашкин С.А.* Архитектурные принципы построения конвейеров машинного обучения для решения задачи управления процессом анализа данных дистанционного зондирования Земли // Нелинейный мир. – 2023. – Т. 21, № 3. – С. 27-37.
2. *Лебедев А.С., Марищук Б.В.* Конвейер машинного обучения // Информационные ресурсы и системы в экономике, науке и образовании: сборник статей XIII Международной научно-практической конференции., Пенза, 24–25 апреля 2023 года / Пензенский государственный технологический университет. – Пенза: Автономная некоммерческая научно-образовательная организация «Приволжский Дом знаний», 2023. – С. 66-73.
3. *Шафорост Н.В., Власов Д.В.* Развертывание модели машинного обучения с использованием современного фреймворка FastAPI // Современное образование: традиции и инновации. – 2023. – № 3. – С. 102-106.
4. *Федоренко М.А.* Автоматизация проверки и уточнения ML-моделей при работе с аудиоданными сосудистого доступа // Наука и инновации XXI века: Сборник статей по материалам VIII Всероссийской конференции молодых ученых. В 4-х томах, Сургут, 23 декабря 2021 года. Том I. – Сургут: Сургутский государственный университет, 2022. – С. 158-160.
5. *Ямиков Р.Р., Григорян К.А.* Анализ и разработка конвейера MLOps для развертывания моделей машинного обучения // Электронные библиотеки. – 2022. – Т. 25, № 2. – С. 177-196.
6. *Чекмарев М.А., Клюев С.Г., Бобров Н.Д.* Анализ методов обеспечения безопасности систем машинного обучения // Воронежский институт высоких технологий, 2022. Оптимизация и информационные технологии. с. 67.
7. *Hapke H., Nelson C.* Building Machine Learning Pipelines. O'Reilly Media, Inc., 2020. 367 p.
8. *Shaikh S.* An End-To-End Machine Learning Pipeline That Ensures Fairness Policies: arXiv:1710.06876. arXiv, 2017.