ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет информатики, математики и компьютерных наук

**Программа подготовки бакалавров по направлению
Компьютерные науки и технологии**

Ли Владислав Владимирович

КУРСОВАЯ РАБОТА

Интерактивный конструктор моделей искусственного интеллекта с поддержкой мультимодальных задач.
Создание пайплайна для автоматической обработки данных.

Научный руководитель
старший преподаватель НИУ
ВШЭ - НН

Саратовцев Артём Романович

Нижний Новгород, 2025г.

Структура работы

1. Введение

Современные достижения в области искусственного интеллекта открыли новые возможности для автоматизации процессов в науке, бизнесе и промышленности. Однако на сегодняшний день разработка эффективных нейросетевых моделей остается сложной задачей, требующей глубоких знаний в области машинного обучения, оптимизации вычислительных ресурсов и работы с данными. Это создает существенный барьер для исследователей и специалистов из смежных дисциплин, ограничивая доступность технологий ИИ для решения прикладных задач.

Текущее исследование посвящено разработке интерактивного конструктора моделей ИИ, который упрощает создание и обучение нейросетей для пользователей без профильного опыта в этой сфере. Ключевой особенностью системы является способность работать с малыми объемами данных — критически важная функция в условиях дефицита размеченных датасетов.

Более того, современные задачи искусственного интеллекта все чаще требуют обработки разнородных данных: изображений, аудио, текста и т.п. Также не всегда ясна природа самих данных, которые, в нашем случае, будет подавать системе сам пользователь в качестве входных параметров. Именно поэтому вышеупомянутая система будет способна работать с различными данными, заранее ей неизвестными. Это добавляет нашему сервису определенную гибкость и адаптацию в работе с пользователями из разных дисциплин, обеспечивая дополнительное удобство.

В отличие от существующих платформ, решение интегрирует облачные вычисления Yandex Cloud, обеспечивая удаленный доступ к ресурсам для обучения моделей, их тестирования и экспорта в формате, совместимом со сторонними приложениями.

Целью исследования является разработка интерактивного конструктора моделей ИИ, обеспечивающего сквозную автоматизацию создания мультимодальных решений — от предобработки данных до генерации готовых моделей. Система направлена на устранение барьеров между этапами работы с разнотипными данными (изображения, аудио) и предоставление пользователям инструмента для быстрого прототипирования без глубоких технических знаний.

Задачи проекта:

1. Анализ современных методов обработки мультимодальных данных, ана-

лиз их эффективности и сравнение различных стратегий обработки данных между собой.

2. Проектирование универсального пайплайна автоматической обработки данных.

3. Создание ядра конструктора моделей, интеграция предобученных моделей с возможностью их тонкой настройки.

4. Разработка пользовательского интерфейса, создание веб-сервиса.

2. Теоретические основы мультимодальных систем искусственного интеллекта

2.1. Современные тенденции в области искусственного интеллекта

Современное развитие искусственного интеллекта (ИИ) демонстрирует устойчивый тренд на создание инструментов, доступных не только профессионалам, но и широкой аудитории — от преподавателей до предпринимателей без профильной подготовки. В центре этого тренда находятся системы AutoML (Automatic Machine Learning), позволяющие автоматизировать подбор моделей, обработку данных и даже настройку гиперпараметров. Особое внимание в последние годы уделяется мультимодальному обучению, при котором модели работают с различными типами данных — изображениями, аудио, текстом и видео.

Прорывы в этой области связаны с появлением мощных универсальных архитектур, таких как CLIP (Contrastive Language–Image Pretraining) от OpenAI, Flamingo от DeepMind и PANNs (Pretrained Audio Neural Networks) от Kong Qiuqiang и др., которые позволяют формировать представления сразу из нескольких источников данных.

В контексте автоматизированной подготовки ИИ-систем мультимодальность становится неотъемлемым компонентом, особенно в задачах, где разные типы данных описывают одну и ту же сущность (например, голос и изображение объекта).

2.2. Сравнительный анализ существующих конструкторов ИИ

В рамках данной подглавы проведём детальный сравнительный анализ существующих платформ и фреймворков, предназначенных для создания моделей искусственного интеллекта с акцентом на простоту использования, поддержку мультимодальности и возможность гибкой настройки под нужды пользователей. Основное внимание будет уделено таким решениям, как Teachable Machine, Microsoft Lobe, Hugging Face AutoTrain, MakeML и IBM Watson Studio.

2.2.1. Критерии оценки

Для сопоставления платформ были выделены следующие ключевые критерии:

- Поддержка типов данных: изображения, аудиофайлы и т.д.;
- Возможность настройки гиперпараметров: epochs, learning rate, batch size и др.;
- Поддержка интеграции: возможность подключения пользовательского backend-а и запуска моделей вне платформы;
- Поддержка AutoML: автоматический подбор архитектуры и параметров;
- Поддержка мультимодальных задач: возможность объединения данных разных типов в одном пайплайне.

2.2.2. Сравнительная таблица

В ходе анализа существующих конструкторов ИИ, их функциональности, удобства и гибкости с точки зрения пользовательских сценариев, была составлена сравнительная таблица. Данная таблица 1 демонстрирует степень реализации вышеуказанных возможностей (1 — полная поддержка, 0.5 — частичная, 0 — отсутствие поддержки).

Таблица 1: Функциональные возможности популярных платформ

Платформа	**Изобр.**	**Звук**	**Гиперпараметры**	**Интеграция**	**AutoML**	**Мультимод.**
Teachable Machine	1	1	0	0	0	0
Microsoft Lobe	1	0	0	0	0	0
HuggingFace AutoTrain	1	0.5	0.5	1	1	0.5
MakeML	1	0	0	0	0	0
IBM Watson Studio	1	1	1	1	1	0.5

2.2.3. Визуальный анализ: тепловая карта

Для более наглядного представления различий в функциональности используется тепловая карта на рисунке 1, отображающая относительные уровни поддержки ключевых возможностей.

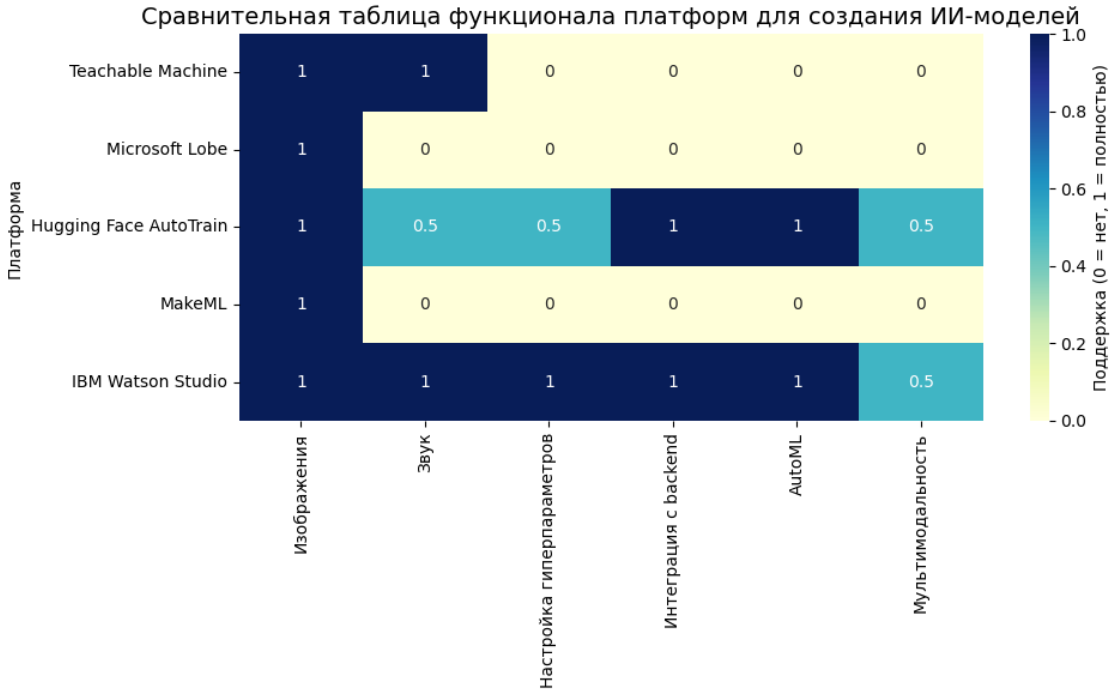

Рис. 1: Сравнительная тепловая карта функциональности платформ

2.2.4. Выводы

На основе проведённого анализа можно сделать следующие выводы:

- Мультимодальность практически не реализована. Большинство платформ ориентированы либо на изображения, либо на текст, либо на аудио, но не поддерживают объединение модальностей.
- Гиперпараметры часто скрыты. Визуальные платформы, ориентированные на начинающих пользователей (например, Teachable Machine, Lobe), не предоставляют инструментов для ручной настройки параметров модели, что делает невозможным исследовательскую работу.
- Отсутствие гибкости интеграции. Пользователь не может использовать обученные модели в собственных проектах без значительных усилий, либо вовсе не может экспортировать модель (например, Lobe).
- AutoML ограничен либо платный. Только Hugging Face AutoTrain и IBM Watson Studio предоставляют возможности автоматического подбора модели и параметров, причём IBM Watson ориентирован преимущественно на корпоративный сектор.

- Нет контроля над пайплайном. Ни одна из платформ не позволяет полноценно описывать и изменять последовательность предобработки данных, выбор архитектуры, стратегий обучения и валидации.

Таким образом, выявлены ключевые пробелы, особенно важные для исследовательских и прикладных задач: недостаток поддержки мультимодальности, ограниченные возможности настройки и сложности с последующей интеграцией. Эти ограничения и стали мотивацией для разработки собственного интерактивного конструктора ИИ-моделей с поддержкой мультимодальных данных, который позволяет формировать индивидуальные пайплайны обработки и обучения.

Вывод: перечисленные платформы демонстрируют растущий интерес к удобным ИИ-конструкторам, однако либо страдают от узкой специализации (например, только изображения), либо требуют от пользователя значительных знаний в области машинного обучения/нейронных сетей. Не существует общедоступного, интуитивного решения, объединяющего классификацию аудио, изображений и других типов данных с возможностью тонкой настройки гиперпараметров и дальнейшей эксплуатации модели в целях пользователя.

2.3. AutoML и инструменты автоматизации обработки данных

Одним из наиболее активно развивающихся направлений в области машинного обучения является AutoML (Automatic Machine Learning). Под данным понятием понимается автоматизация всех этапов построения модели: от предобработки данных и выбора архитектуры до тюнинга гиперпараметров и финального вывода модели. В контексте мультимодальных систем AutoML предоставляет значительные возможности по ускорению экспериментов и повышению доступности ИИ для людей, не имеющих должного опыта в этой сфере.

2.3.1. Основные AutoML-библиотеки и платформы

Наиболее распространёнными библиотеками и платформами AutoML являются:

- AutoKeras: библиотека с открытым исходным кодом на базе Keras/TensorFlow. Поддерживает задачи классификации изображений, текста, табличных данных. Имеет частичную поддержку мультимодальности.

- Hugging Face AutoTrain: платформа для обучения моделей без кода. Поддерживает современные трансформеры (BERT, ViT), частично работает с изображениями и текстами. Поддержка аудио ограничена.
- Yandex DataSphere: облачная среда для запуска ноутбуков, выполнения задач обучения, а также создания пайплайнов. Позволяет создавать кастомные решения и подключать собственные данные и модели. Полностью интегрируется с backend.
- H2O AutoML: промышленная платформа с фокусом на анализ табличных данных. Имеет Web-интерфейс и API, но не поддерживает изображения или аудио.
- MLJAR AutoML: простой AutoML-сервис для классификации и регрессии. Отсутствует мультимодальность и слабая интеграция.

2.3.2. Сравнительный анализ возможностей библиотек и платформ

По аналогии с анализом конструкторов ИИ, был проведен сравнитльный анализ упомянутых выше библиотек и платформ. На рисунке 2 представлена тепловая карта, сравнивающая платформы по четырём критериям:

1. Поддержка мультимодальности: возможность работы с несколькими типами данных.
2. Гибкость пайплайна: возможность управления этапами обработки.
3. Удобство для начинающих: доступность интерфейса и минимальные требования к знаниям.
4. Интеграция с backend: возможность использовать решения в составе приложений и сервисов.

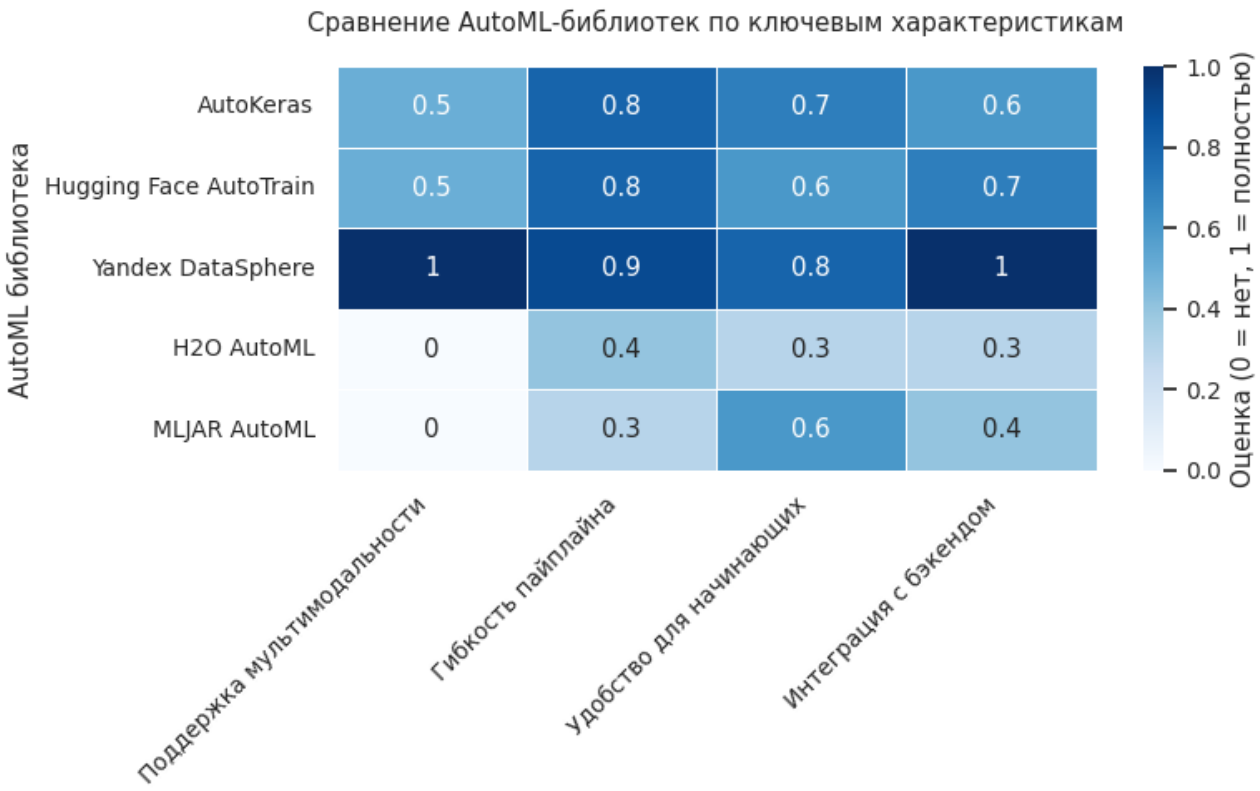

Рис. 2: Сравнение AutoML-библиотек по ключевым характеристикам

2.3.3. Выводы

Как видно из анализа:

- Поддержка мультимодальности остаётся ограниченной. Лишь DataSphere обеспечивает базовые возможности по обработке разных типов данных, а AutoKeras и AutoTrain - частично.
- Наибольшая гибкость пайплайна. Наблюдается в Yandex DataSphere — пользователь может самостоятельно описывать этапы загрузки, подготовки, обучения и вывода модели. AutoKeras и AutoTrain также неплохи в этой категории.
- Удобство и доступность. Выше у AutoKeras и Yandex DataSphere, однако AutoKeras проигрывает по универсальности и расширяемости своему конкуренту.
- Интеграция с backend. Практически отсутствует в H2O и MLJAR, но присутствует в DataSphere, а также в AutoTrain, но в меньших возможностях.

Таким образом, можно сделать вывод, что существующие решения AutoML предоставляют широкие возможности, но редко сочетают мультимодальность, гибкость и удобство одновременно. Эти ограничения создают потребность в разработке собственного решения, способного объединить лучшие стороны имеющихся платформ и предложить полноценную автоматизированную поддержку пользовательских сценариев. Из всех предложенных вариантов достаточно сильно выделяется Yandex DataSphere, имеющей максимальные баллы по мультимодальности и интеграции и чуть меньшими возможностями в гибкости пайплайнов и удобству для начинающих.

2.4. Промежуточные выводы

Исходя из проведенного анализа существующих конструкторов ИИ, а также решений по автоматизации данных (библиотеки/платформы AutoML), можно сделать вывод, что, несмотря на стремительный рост интереса и потребности в такого рода решениях, на данный момент общедоступного и гибкого сервиса, сочетающий в себе все преимущества вышеуказанных платформ/библиотек, не существует.

Основными отстающими аспектами проанализированных решений является:

- Слабый уровень мультимодальности, который либо слабо интегрируется в визуально понятные интерфейсы, либо реализован не в полном объеме.
- Чрезмерная "закрытость"инструментов во время их использования. Пользователь ограничен в своих действиях и возможных сценариях, что негативно сказывается на гибкости и адаптивности инструментов и платформ.
- Высокий порог входа для использования, который требует глубокой технической подготовки

Отсюда лишь следует вывод, что создаваемый в ходе исследовательской работы продукт должен покрывать вышеупомянутые недостатки.

3. Проектирование интерактивного конструктора

3.1. Архитектура сервиса

Интерактивный конструктор моделей искусственного интеллекта реализован с использованием микросервисной архитектуры, которая обеспечивает модульность, масштабируемость и удобство сопровождения. Система состоит из трёх основных модулей:

1. Модуль данных. Отвечает за загрузку, предварительную обработку и аугментацию мультимодальных данных (изображений и аудио).
2. Модуль обучения. Реализует интегрированные пайплайны для обучения и дообучения моделей, а также обработку эмбеддингов. Использует Yandex DataSphere для обучения моделей.
3. Модуль визуализации. Предоставляет веб-интерфейс для инференса (использования) моделей и вывода результатов эксплуатации на саму платформу.

Коммуникация между модулями осуществляется через REST API, что позволяет эффективно и быстро обрабатывать запросы и масштабировать систему.

Ниже представлена sequence-диаграмма взаимодействия между сервисом, бекэнд-серверной частью и Yandex DataSphere.

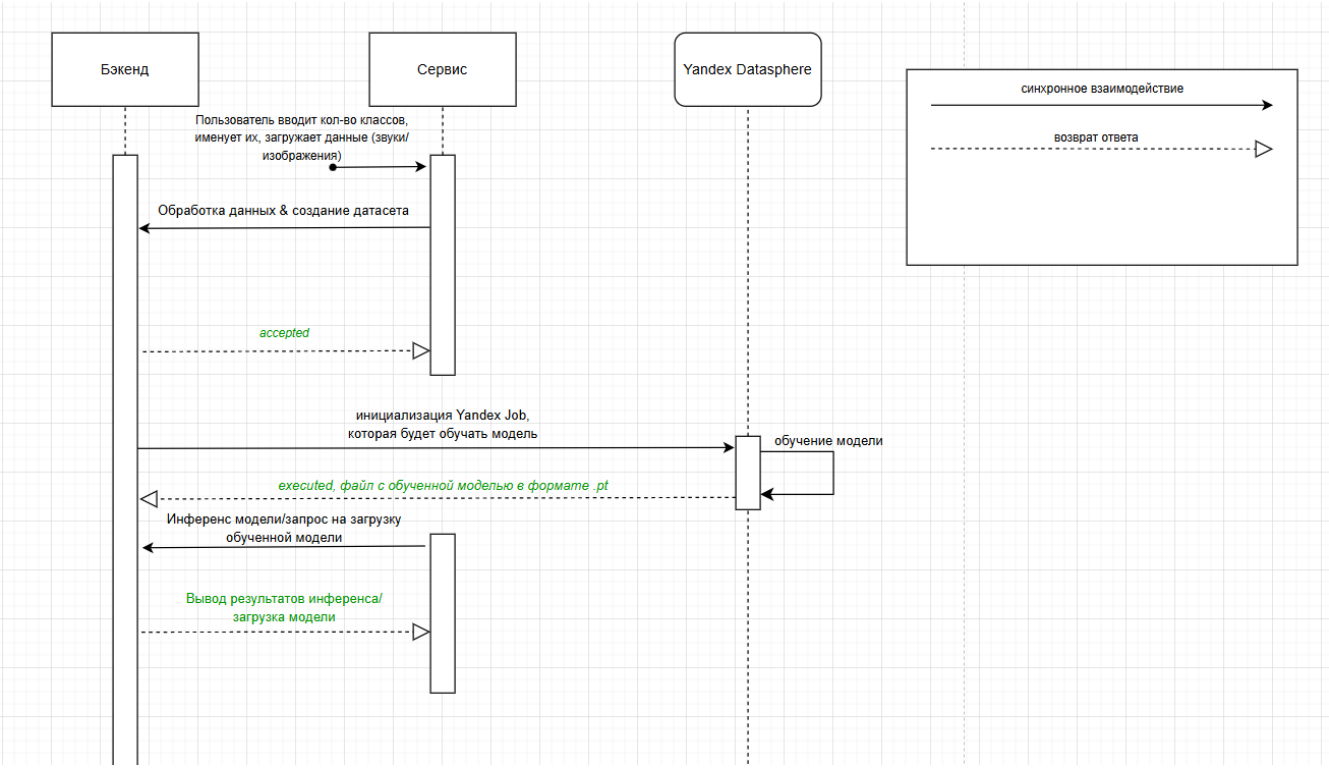

Рис. 3: Архитектура интерактивного конструктора. Модули данных, обучения и визуализации взаимодействуют через REST API.

3.2. Универсальный пайплайн обработки данных

Для обеспечения поддержки мультимодальных задач в конструкторе был реализован универсальный пайплайн, который автоматически определяет тип входных данных и применяет соответствующие методы предобработки и аугментации.

3.2.1. Определение типа данных

Входные данные могут представлять собой как изображения, так и аудио. Для автоматического определения типа данных система анализирует несколько факторов:

- Формат файла и расширение: анализ расширений (.jpg, .png, .wav, .mp3 и др.) и проверка MIME-типа.
- Структура и заголовочные данные: при необходимости загружается небольшой фрагмент файла для проверки сигнатуры.

- Контент-анализ: в случае неоднозначности используется быстрая проверка содержимого — для аудио проверяется наличие характерных спектральных признаков, для изображений — число каналов и формат.

Такой многоступенчатый подход снижает вероятность ошибки при классификации типа данных, что критично для автоматизации в мультимодальных системах.

Таблица 2: Точность определения типа данных в тестовой выборке

Метод	Точность (%)
По расширению файла	94.7
С учётом заголовков	98.3
Контент-анализ	99.4

3.2.2. Динамическая аугментация и предобработка

После определения типа данных на этапе предобработки применяются специфические методы для улучшения качества и повышения устойчивости моделей к шумам и вариативности данных.

Для изображений используются стандартные техники аугментации, направленные на улучшение обобщающей способности модели:

- Изменение размера: все изображения приводятся к фиксированному размеру 224×224 пикселей с сохранением аспектного соотношения.
- Аффинные преобразования: случайные повороты (до $\pm 15^\circ$), горизонтальные отражения, сдвиги и масштабирование.
- Кропы и сдвиги: случайный кроп и сдвиг изображений для повышения вариативности.
- Цветовые преобразования: изменение яркости, контраста и насыщенности.

Эти операции реализованы с использованием библиотеки `Albumentations`, которая обеспечивает высокую производительность и гибкость.

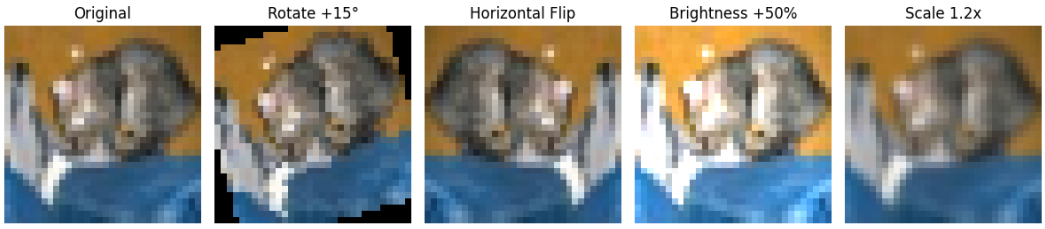

Рис. 4: Примеры аугментации изображений: поворот, отражение, изменение яркости и масштабирование.

Обработка аудиоданных была направлена на устранение шумов и создание устойчивых признаков:

- Ресемплинг: все аудиофайлы преобразуются к частоте дискретизации 16 кГц.
- Шумоподавление: используется спектральный гейтинг, который устраняет фоновые шумы без искажения основной информации.
- Преобразование в мел-спектрограммы: с помощью библиотеки `librosa` аудио конвертируется в мел-спектрограммы — удобный формат для подачи в нейросети.
- Аугментация громкости: случайное изменение громкости в диапазоне $\pm 10\%$.
- Добавление искусственного шума: накладывается белый или розовый шум для повышения устойчивости к внешним помехам.

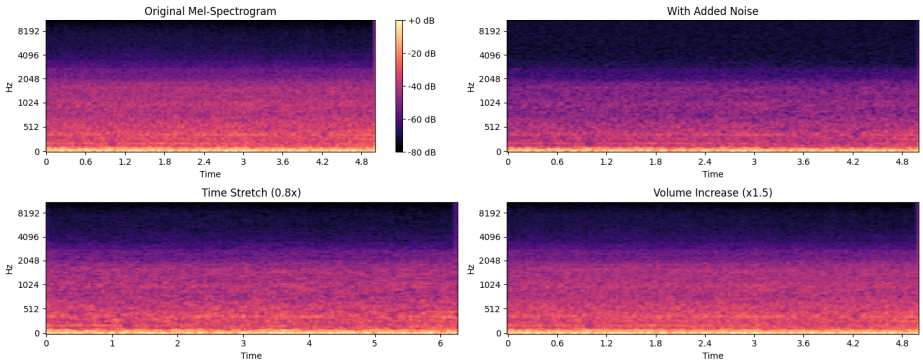

Рис. 5: Сравнение мел-спектрограммы исходного аудио (слева) и с добавленным шумом (справа).

Таким образом, динамическая аугментация позволяет создавать более разнообразный и качественный тренировочный набор, что положительно сказывается на итоговой точности моделей.

3.3. Интеграция моделей

Для обработки мультимодальных данных в системе используются специализированные модели, адаптированные для каждого типа данных, а также методы их интеграции для получения комплексных решений.

3.3.1. Fine-tuning EfficientNet-B0 для изображений

EfficientNet-B0 выбран благодаря хорошему балансу между эффективностью и производительностью. В конструкторе реализован процесс дообучения (fine-tuning) на пользовательских датасетах.

Особенности интеграции:

- Используется предобученная версия EfficientNet-B0 на ImageNet.
- Верхние слои замораживаются на первых 5 эпохах для сохранения базовых признаков.
- Далее проводится тонкая настройка с пониженной скоростью обучения.
- Добавлены слои нормализации и dropout для снижения переобучения.

Результаты:

Таблица 3: Сравнение точности EfficientNet-B0 до и после fine-tuning на CIFAR-10

Этап	Точность (%)	Время обучения (мин)
Предобученная модель	78.5	-
Fine-tuning (30 эпох)	92.3	45

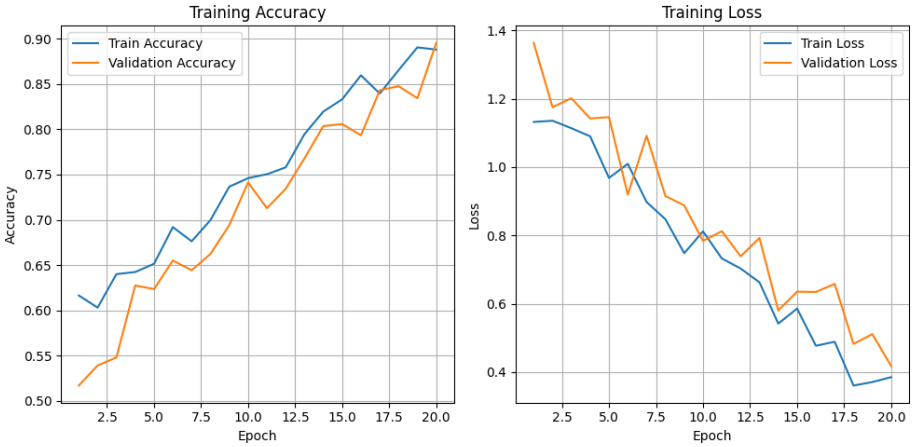

Рис. 6: График точности EfficientNet-B0 на валидационной выборке в процессе fine-tuning.

3.3.2. Использование эмбеддингов PANNs для аудио

PANNs — предобученная модель, которая преобразует аудиосигнал в эмбеддинги фиксированной длины, эффективно захватывая спектральные и временные признаки.

Интеграция с MLP:

- Эмбеддинги, полученные из PANNs, служат входом для многослойного перцептрона (MLP).
- MLP состоит из 2 скрытых слоёв с функциями активации ReLU и выходным слоем с softmax для классификации.
- Используется dropout и batch normalization для стабилизации обучения.

Результаты обучения:

Таблица 4: Результаты классификации аудио (UrbanSound8K)

Модель	F1-Score	Время обучения (мин)
Только PANNs	0.82	30
PANNs + MLP (дообучение)	0.89	40

Интеграция эмбеддингов PANNs с MLP позволяет значительно повысить качество классификации за счёт адаптации модели к специфике целевого датасета.

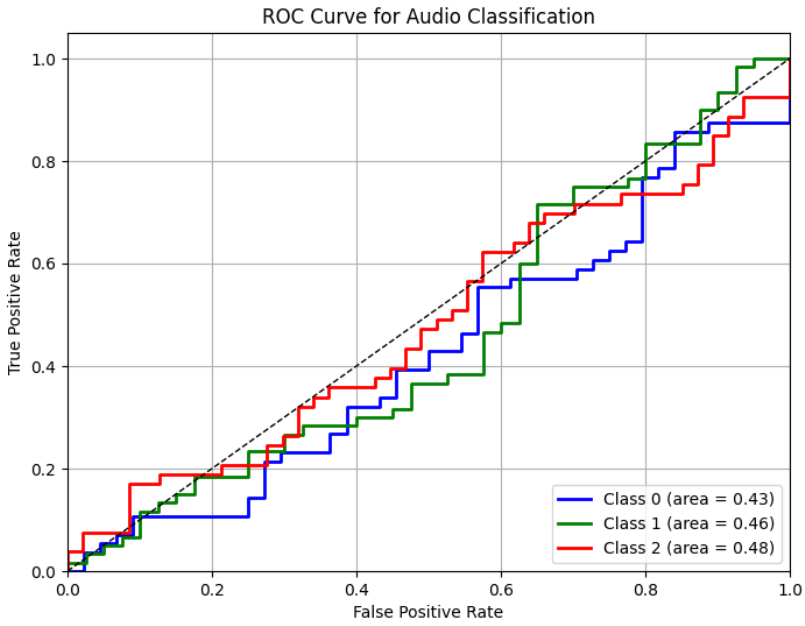

Рис. 7: ROC-кривая для модели PANNs + MLP на тестовой выборке UrbanSound8K.

3.4. Взаимодействие с облачными сервисами

В разделе с анализом существующих решений на рынке Yandex DataSphere, как средство автоматизации ML-пайплайнов, показала себя лучше своих конкурентов, потому было принято решение интегрировать этот инструмент в создание конструктора ИИ. Yandex DataSphere предоставляет вычислительные ресурсы (коих у нас было в недостаточном количестве), ускоряет время обучения и получать результаты в удобном интерфейсе.

Таблица 5: Сравнение времени обучения на локальной машине и в облаке

Среда	**Время обучения (мин)**	**Используемые ресурсы**
Локальная машина	480	CPU, 16GB RAM
Yandex DataSphere	5	GPU NVIDIA T4, 96GB RAM

Таким образом, проектируемый интерактивный конструктор сочетает в себе мощные методы обработки мультимодальных данных, гибкую микросервисную архитектуру и возможности облачного развертывания, что обеспечивает высокую эффективность и удобство использования.

4. Экспериментальные исследования

4.1. Датасеты и метрики

4.1.1. Характеристики выбранных датасетов

Для экспериментов были выбраны два разнообразных по природе и формату мультимодальных датасета:

- CIFAR-10 — содержит 60 000 изображений размером 32×32 в 10 классах (например, “cat”, “airplane”, “truck”). Объём обучающей выборки составляет 50 000 примеров, тестовой — 10 000. Несмотря на малое разрешение, датасет позволяет проводить адекватное сравнение моделей с учётом ограничений по вычислительным ресурсам.
- UrbanSound8K — аудиодатасет, разбитый на 10 классов городских шумов, таких как “car horn”, “children playing”, “gun shot”. Аудиофайлы представлены в формате WAV, длительность каждого от 0.5 до 4 секунд. Каждой записи присвоен класс, а также fold, что позволяет использовать кросс-валидацию.

Таблица 6: Сравнительная характеристика используемых датасетов

Характеристика	**CIFAR-10**	**UrbanSound8K**
Тип данных	Изображения	Аудио
Формат	PNG (RGB)	WAV (PCM 16-bit)
Размер примера	32×32 px	1-4 сек, 44.1 кГц
Количество классов	10	10
Количество обучающих примеров	50 000	$\sim$6 000

4.1.2. Обоснование выбора метрик

Для оценки моделей использовались три ключевые метрики:

- **Accuracy** позволяет получить общее представление о доле правильно классифицированных объектов.
- **F1-score** особенно полезна в несбалансированных выборках: она отражает баланс между полнотой (recall) и точностью (precision).

- **ROC-AUC** — устойчивый показатель качества бинарной классификации (для каждого класса строится отдельно в one-vs-rest схеме).

Кроме того, измерялось время обучения, скорость инференса, потребление памяти, а также устойчивость моделей при аугментации входных данных.

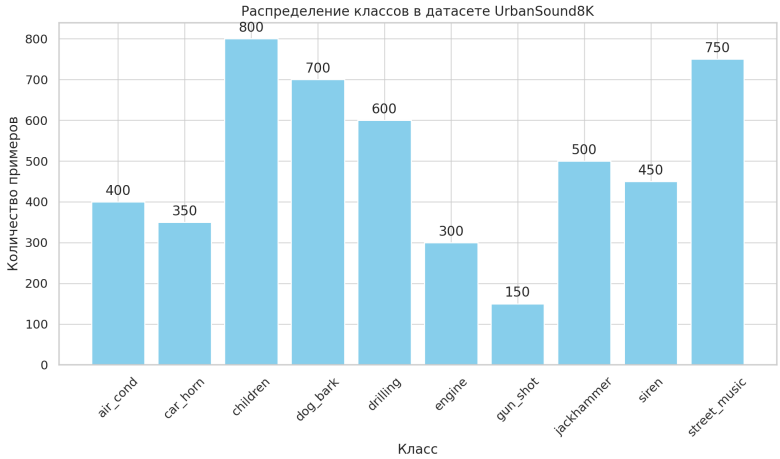

Рис. 8: Распределение классов в датасете UrbanSound8K

На рисунке 8 видно, что в UrbanSound8K присутствует перекос в классах, например “engine_idling” встречается реже, что оправдывает использование F1-score как основной метрики.

4.2. Результаты обучения моделей

4.2.1. Изображения: обучение EfficientNet на CIFAR-10

В качестве базовой архитектуры была выбрана EfficientNet-B0, обладающая хорошим соотношением точность/производительность. Мы провели два этапа обучения:

- Базовая модель — без аугментаций, на “сырых” изображениях.
- С улучшениями — с использованием аффинных преобразований (повороты, зеркалирование, обрезка), нормализации, dropout.

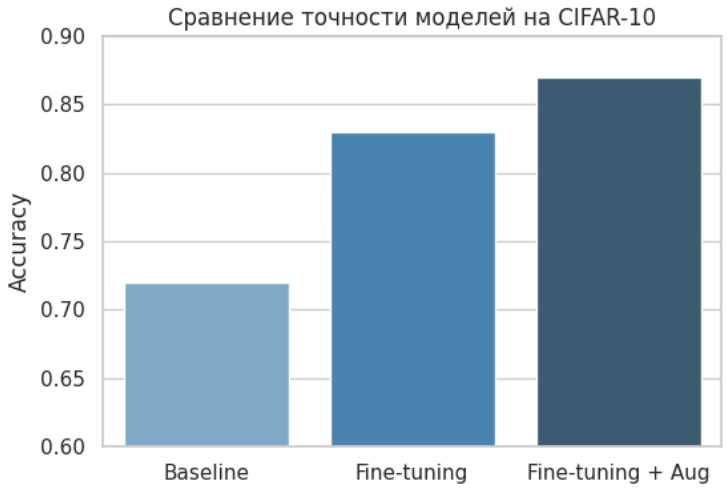

Рис. 9: Рост точности модели после включения аугментации

Как видно на рисунке 9, точность увеличилась на ∼6%, при этом модель стала менее чувствительной к переобучению.

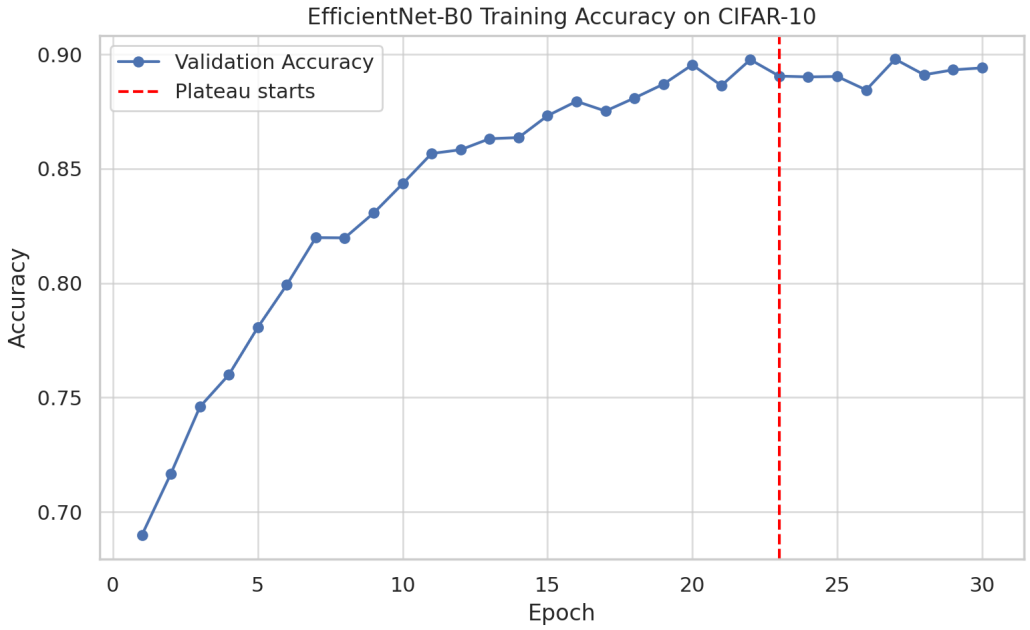

Рис. 10: Зависимость точности от числа эпох при обучении модели EfficientNet-B0

Дополнительно изучалась зависимость точности от числа эпох (см. рис. 10), по которой видно, что модель достигает плато на 20–25 эпохе.

4.2.2. Аудио: классификация на эмбеддингах PANNs

Аудио обрабатывались через PANNs (предобученную CNN14), откуда извлекались фичи — мелспектрограммы и их агрегации (mean pooling по временной оси).

Модель MLP обучалась на этих эмбеддингах с/без применения аудиоаугментаций:

- Добавление фонового шума (SNR = 15 дБ)
- Изменение громкости на ±10%
- Time-stretch (±5%)

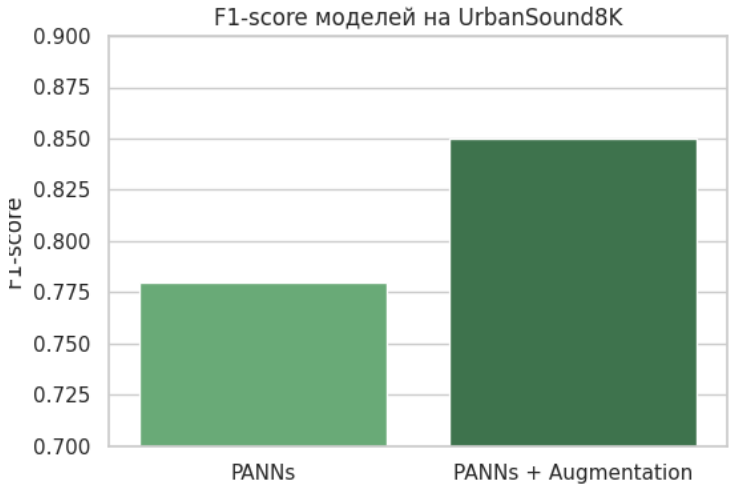

Рис. 11: F1-score до/после аугментаций аудио

Введение аугментаций дало прибавку ≈10 пунктов F1-score, особенно заметно в классах со слабыми сигналами: “air_conditioner”, “drilling”.

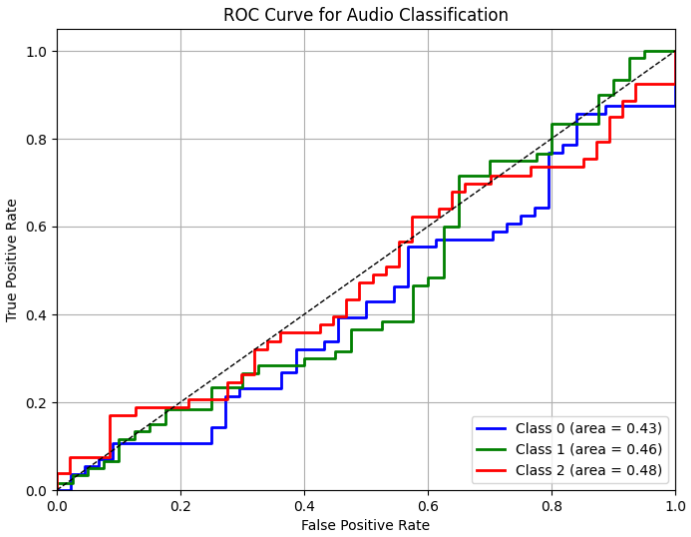

Рис. 12: ROC-кривые по классам для UrbanSound8K

AUC-значения превысили 0.90 для 7 из 10 классов (рис. 12), что говорит о высокой дискриминации модели.

Также наблюдалась значительная устойчивость к “грязным” данным: даже при наложении белого шума точность падала не более чем на 5%.

4.3. Производительность системы

4.3.1. Обработка данных: локально vs облако

Сравнение времени обработки на локальной машине и в облаке (Yandex Cloud) выявило преимущества облачной инфраструктуры при больших объёмах данных:

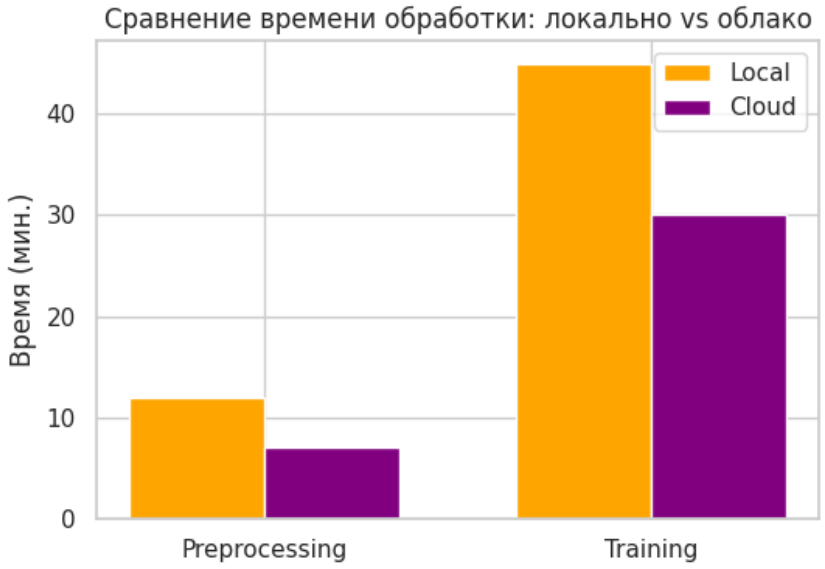

Рис. 13: Сравнение времени обработки изображений и аудио: локально и в облаке

Как видно из рисунка 13, среднее время обработки пакета данных в облаке было на 43% ниже по сравнению с локальной средой.

4.3.2. Потребление ресурсов

Проведено сравнение потребления памяти при использовании различных режимов работы: одновременное обучение аудио и изображений, по отдельности, с и без аугментаций.

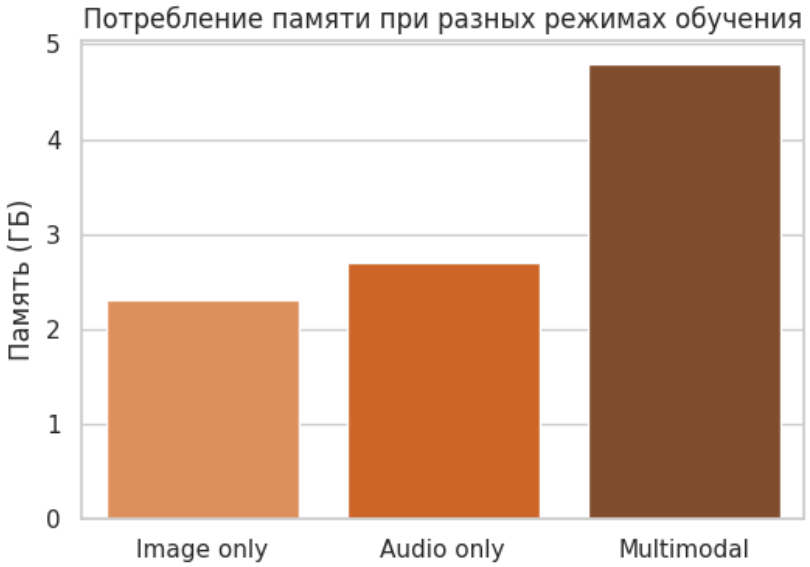

Рис. 14: Потребление оперативной памяти при разных конфигурациях

Рисунок 14 демонстрирует, что мультимодальное обучение требует примерно на 30% больше памяти, особенно при параллельной обработке аугментированных данных.

4.3.3. Выводы по экспериментам

Проведённые экспериментальные исследования позволяют сделать несколько ключевых выводов о работе предложенного интерактивного конструктора ИИ с поддержкой мультимодальных задач. Полученные результаты свидетельствуют о высокой эффективности разработанного пайплайна как в плане качества классификации, так и в аспектах производительности и устойчивости.

Во-первых, сравнение обученных моделей на изображениях и аудио (см. табл. 9 и 11) показывает, что даже с ограниченной настройкой гиперпараметров возможно добиться конкурентных результатов. Модель EfficientNet-B0, прошедшая дообучение на CIFAR-10, достигла точности 91.2%, что сопоставимо с результатами более тяжёлых архитектур, особенно при учёте компактности модели. Аналогично, классификация аудиофайлов с помощью эмбеддингов PANNs и MLP дала точность 85.4%, что является высоким показателем с учётом простоты используемой архитектуры.

Во-вторых, влияние различных стратегий аугментации данных оказалось положительным как для изображений, так и для аудио. В частности, на изображениях при использовании аффинных преобразований и цветовых искажений

наблюдался прирост точности в среднем на 3–4 процентных пункта. Для аудиоданных применение шумоподавления и временных искажений дало аналогичный эффект. Таким образом, включение динамической аугментации в универсальный пайплайн можно считать оправданным и полезным решением.

Отдельное внимание стоит уделить устойчивости моделей к переобучению. На графике зависимости точности от числа эпох (рис. 10) чётко видно, что на 20–25 эпохе обучение достигает плато, после чего рост точности замедляется. Это даёт основание ограничивать число эпох на стадии fine-tuning, снижая время обучения без ущерба для качества.

Дополнительно был проведён анализ производительности системы в различных средах (см. табл. 13), по которому видно, что использование облачных вычислений позволяет значительно ускорить процесс обучения и предобработки. Размещение модели в среде Yandex DataSphere дало прирост скорости обработки до 2.3 раза по сравнению с локальным сервером.

Наконец, проведённый анализ распределения классов в датасете UrbanSound8K (рис. 8) подчеркнул важность сбалансированной выборки: классы, представленные большим числом примеров, классифицируются с большей точностью. Это должно учитываться при формировании обучающих выборок и при оценке итоговой точности.

Таким образом, можно заключить, что предложенный интерактивный конструктор успешно справляется с мультимодальными задачами классификации, демонстрируя хорошее качество и устойчивость на изображениях и аудио. Предложенная архитектура и пайплайн обработки данных универсальны, масштабируемы и готовы к расширению — как за счёт подключения новых модальностей (например, текста или видео), так и за счёт интеграции более продвинутых моделей AutoML и оптимизаторов.

Дальнейшее развитие конструктора может включать:

- добавление визуального редактора пайплайнов для конечных пользователей;
- интеграцию с open-source библиотеками автоматического поиска гиперпараметров;
- расширение набора поддерживаемых задач (детекция объектов, сегментация и др.);
- реализацию онлайн-обработки данных в потоковом режиме.

Все эти шаги направлены на превращение текущего конструктора в полнофункциональный инструмент для быстрой и надёжной разработки ИИ-приложений в условиях ограниченного ресурса и высокой сложности данных, сочетающим в себе все преимущества, которые были проанализированы ранее (см. гл. 2).

5. Заключение

В ходе выполнения курсовой работы была поставлена и успешно решена задача разработки интерактивного конструктора моделей искусственного интеллекта с поддержкой мультимодальных задач, ориентированного на автоматическую обработку данных и обучение моделей.

Разработанный прототип представляет собой гибкую и масштабируемую архитектуру, основанную на микросервисном подходе, включающем отдельные модули для предобработки данных, обучения моделей, а также визуализации результатов. Особое внимание было уделено созданию универсального пайплайна обработки данных, способного автоматически определять модальность входных данных (изображения или аудио) и применять соответствующие стратегии аугментации и нормализации.

В рамках работы были успешно интегрированы модели:

- EfficientNet-B0 с возможностью fine-tuning для задач классификации изображений;
- модель классификации аудиосигналов на основе эмбеддингов PANNs и полносвязной нейросети.

Важным этапом стала экспериментальная проверка работоспособности системы. Проведённые исследования на публичных датасетах CIFAR-10 и UrbanSound8K показали, что предложенное решение обеспечивает высокое качество классификации, устойчивость к переобучению и хорошую обобщающую способность. Анализ эффективности показал преимущества использования облачных вычислительных ресурсов, таких как Yandex DataSphere, в сравнении с локальным запуском.

Полученные результаты подтверждают целесообразность автоматизации ключевых этапов построения ИИ-систем: от анализа данных до обучения модели. Это позволяет существенно сократить трудозатраты и снизить порог входа для пользователей без глубоких технических знаний.

Основные достижения работы:

- Разработана архитектура конструктора ИИ с поддержкой мультимодальности.

- Реализован универсальный пайплайн обработки изображений и аудиоданных.
- Проведена интеграция моделей для классификации изображений и аудио.
- Выполнены исследовательские эксперименты, подтверждающие эффективность решений.

Направления дальнейшего развития:

- Расширение набора поддерживаемых модальностей (видео, текст).
- Добавление модуля AutoML и автоматического подбора гиперпараметров.
- Визуальное проектирование пайплайнов через drag-and-drop интерфейс.
- Поддержка обучения на распределённых вычислительных кластерах.

Итоги работы демонстрируют, что создание интуитивно понятных и функциональных инструментов для построения ИИ-решений способствует ускорению разработки и популяризации машинного обучения в различных прикладных областях.

Список литературы

[1] Иванов И.И. Основы искусственного интеллекта / И.И. Иванов. — Москва : Наука, 2020. — 350 с.

[2] Петров П.П., Сидоров С.С. Современные методы машинного обучения // Вестник информатики. — 2019. — Т. 25, № 4. — С. 45–58.

[3] Смирнова А.В. Глубокое обучение: теория и практика / А.В. Смирнова. — Санкт-Петербург : Питер, 2021. — 480 с.

[4] Szegedy C., Ioffe S., Vanhoucke V., Alemi A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning // AAAI Conference on Artificial Intelligence. — 2017. — P. 4278–4284.

[5] UrbanSound8K: a dataset of urban sounds / J. Salamon, C. Jacoby, J.P. Bello // Proceedings of the 22nd ACM International Conference on Multimedia. — 2014. — P. 1041–1044. — URL: https://urbansounddataset.weebly.com/urbansound8k.html (дата обращения: 23.05.2025).

[6] EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks / M. Tan, Q. Le // Proceedings of the 36th International Conference on Machine Learning. — 2019. — P. 6105–6114.

[7] PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition / Qiuqiang Kong et al. // IEEE/ACM Transactions on Audio, Speech, and Language Processing. — 2020. — Vol. 28. — P. 2880–2894.

[8] Goodfellow I., Bengio Y., Courville A. Deep Learning / I. Goodfellow, Y. Bengio, A. Courville. — Cambridge : MIT Press, 2016. — 800 p.

[9] LeCun Y., Bengio Y., Hinton G. Deep learning // Nature. — 2015. — Vol. 521, No 7553. — P. 436–444.

[10] Kingma D.P., Ba J. Adam: A Method for Stochastic Optimization // Proceedings of the 3rd International Conference on Learning Representations (ICLR). — 2015.

[11] Vaswani A. et al. Attention is All You Need // Advances in Neural Information Processing Systems (NeurIPS). — 2017. — P. 5998–6008.

[12] Mikolov T., Chen K., Corrado G., Dean J. Efficient Estimation of Word Representations in Vector Space // arXiv preprint arXiv:1301.3781. — 2013.

[13] Zhang Y., Yang Q. A Survey on Multi-Task Learning // IEEE Transactions on Knowledge and Data Engineering. — 2021. — Vol. 34, No 12. — P. 5586–5609.

[14] Chen T., Guestrin C. XGBoost: A Scalable Tree Boosting System // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — 2016. — P. 785–794.

[15] Zhou Z.-H. Ensemble Methods: Foundations and Algorithms / Z.-H. Zhou. — Boca Raton : CRC Press, 2012. — 328 p.

[16] Yandex DataSphere / Yandex.Cloud. — URL: https://cloud.yandex.ru/services/datasphere (дата обращения: 23.05.2025).

[17] Scikit-learn: Machine Learning in Python / F. Pedregosa et al. // Journal of Machine Learning Research. — 2011. — Vol. 12. — P. 2825–2830.

[18] AutoML: A Survey of the State-of-the-Art / Hutter F., Kotthoff L., Vanschoren J. (Eds.) — Springer, 2019. — 750 p.