Optimisation of a generative framework for generalist legal contract drafting based on LLM

Romashkina Arina, 224
Strelkova Daria, 223

Project Supervisor: Borevsky Andrey Olegovich

Faculty of Computer Science, HSE University

Legal Tech

- Анализ судебной практики и прогнозирование исхода дел
- Автоматизация работы с документами
- Интеллектуальные чат-боты и виртуальные помощники
- Легальный due diligence и compliance
- Поиск и обработка патентов/интеллектуальной собственности

Постановка задачи

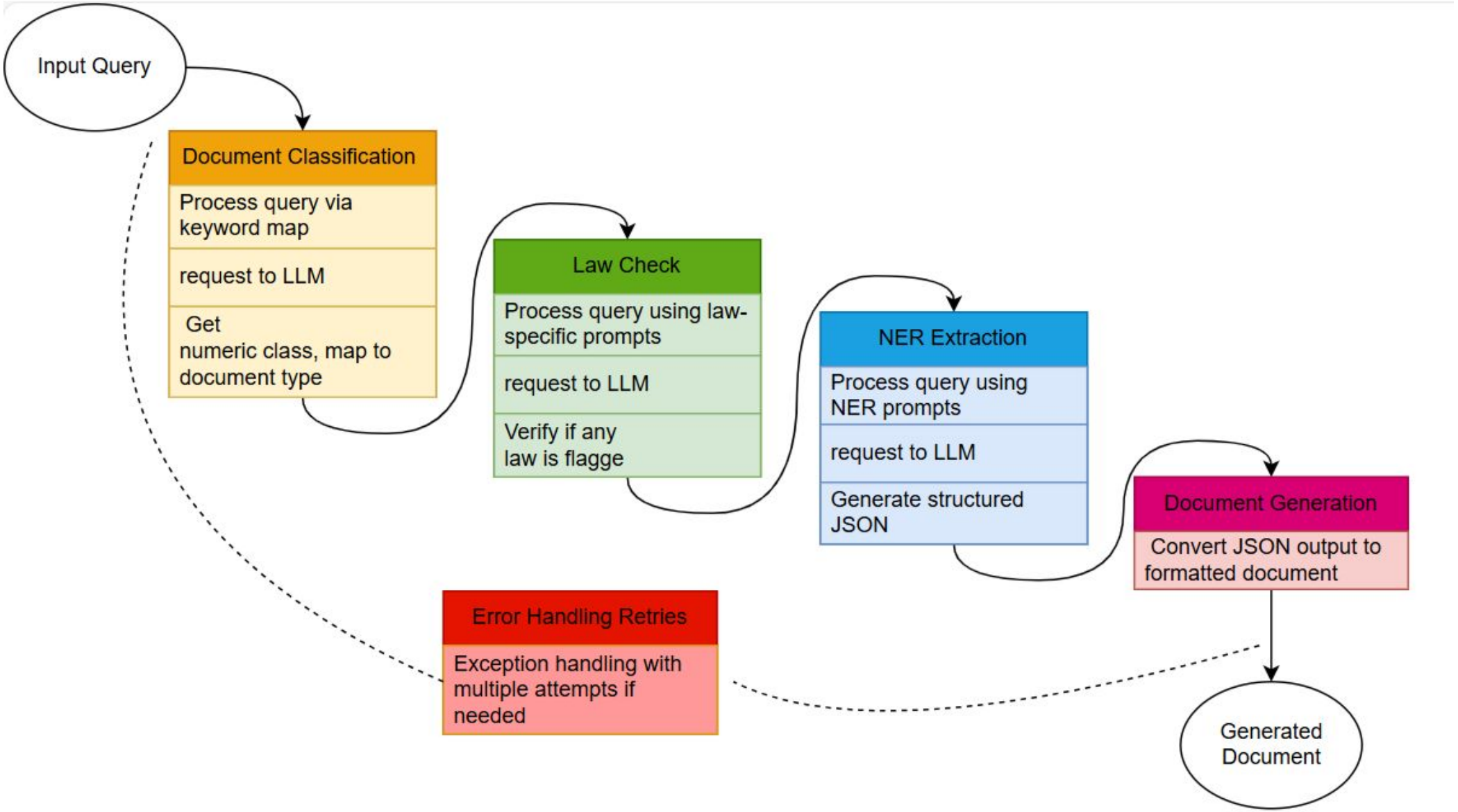
1) **Высокая производительность:** минимизация время отклика системы и эффективная обработка параллельных запросов

2) **Структурированная генерация**: структурированный вывод в формате JSON на каждом этапе

3) **Модульность пайплайна**

4) **Точность классификации**

Общая архитектура системы



Пример запроса для генерации

Нужен договор, по которому ИП Ивлев Александр Петрович ЕГРИП 1234965647362 ИНН 4723943292 купит у индивидуального предпринимателя Шахназарова Сергея Валерьевича ЕГРИП 1234998767362 ИНН 4087643292, комплект летних шин Ikon Tyres (Nokian Tyres) Autograph Aqua 3 225/55 R16 99W XL Размер(Ш/В/Д) 225/55 R16, оплата будет безналичными, цена комплекта составляет 120 000 тысяч рублей, денежные средства должны быть переведены на счет Продавца: р/с 93843746273848 БИК 756462626 в течение 10 рабочих дней с даты настоящего договора. При этом шины должны быть доставлены по адресу Моска, 3-й квартал Капотня, д.26, стр.16А в течение 5 рабочих дней с момента подписания настоящего договора. Неустойка за неисполнение обязательств любой из сторон установлена в размере 3% от суммы сделки за каждый день просрочки исполнения обязательств.

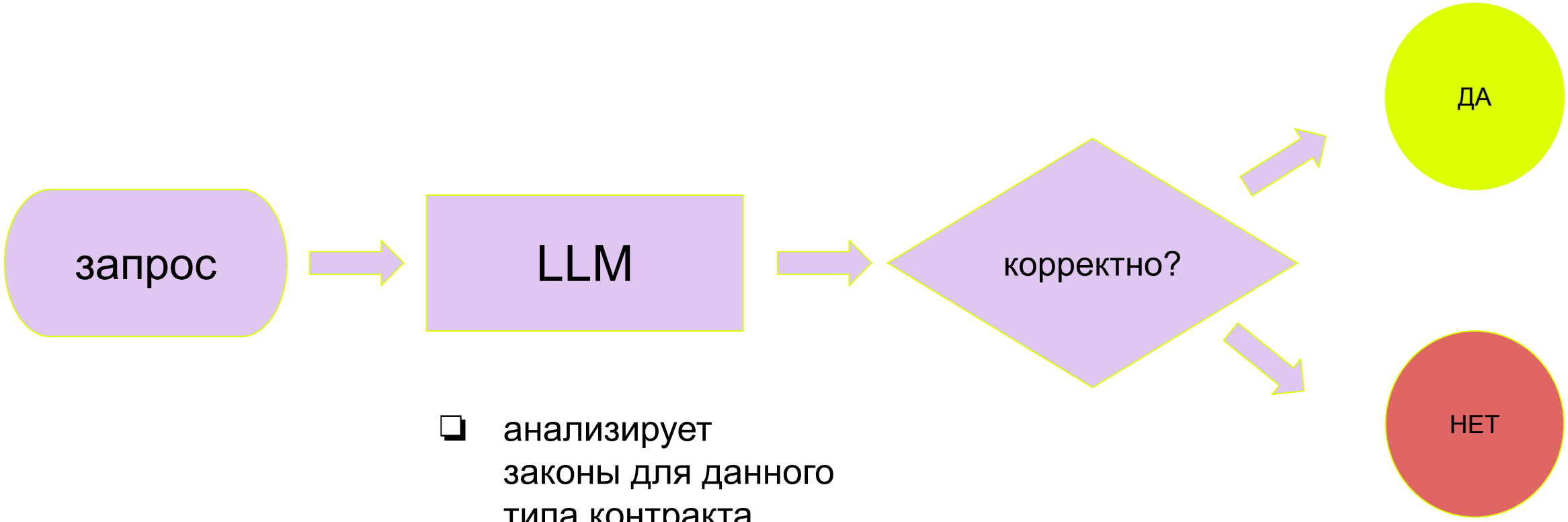
Классификация документов

запрос → LLM → тип договора

- анализирует ключевые слова
- использует признаки, сформированные юристами

- Договор купли-продажи
- Договор оказания услуг
- Договор займа
- Договор дарения
- Договор аренды

Проверка юридической корректности



- анализирует законы для данного типа контракта
- использует признаки, сформированные юристами

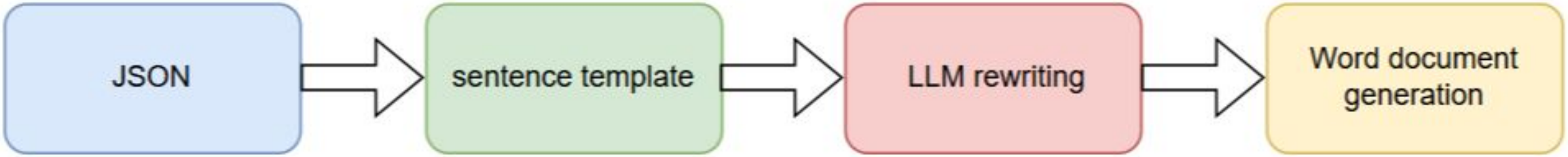
Распознавание именованных сущностей

- находит именованные сущности для данного типа контракта
- вся персональная информация

Пример:

"ИП Сидоров, ИНН 1234567890" → {"name": "ИП Сидоров", "inn": "1234567890"}.

Генерация документов



- Заполняет шаблоны (например, "Договор № __ от __").
- Применяет стили (нумерованные списки, заголовки).
- Инструменты: python-docx.
- Результат: Готовый документ с подставленными данными.

Пример сгенерированного документа

ДОГОВОР КУПЛИ-ПРОДАЖИ ТОВАРА

г. Москва 26.03.2025

Общество с ограниченной ответственностью Невероятные поставки ИНН 7708093971, именуемое в дальнейшем Продавец, с одной стороны, Общество с ограниченной ответственностью Высокое потребление ИНН 7707083893, именуемое в дальнейшем Покупатель, с другой стороны, именуемые вместе Стороны, заключили настоящий Договор (далее Договор) о нижеследующем:

1. ПРЕДМЕТ ДОГОВОРА

1.1. По Договору Продавец обязуется передать Товар в собственность Покупателю, а Покупатель обязуется принять этот Товар и уплатить за него Цену.

1.2. Продавец обязуется передать Покупателю беспроводные наушники Marshall Major IV Black, код товара: 109621 (именуемый в дальнейшем Товар).

1.3. Продавец обязуется обеспечить доставку Товар по адресу фактического местонахождения Покупателя в разумный срок.

1.4. Если обязательства Продавца по передаче Товара были исполнены не в надлежащем виде, то Покупатель вправе отказаться от его исполнения в целом или в части, соответствующей непредставленному исполнению. В случае принятия исполнения в части Договор считается действительным и исполнимым в соответствующей части.

Техническая реализация и эксперименты

- Модель Vikhrmodels/Vikhr-Nemo-12B-Instruct-R-21-09-24 в качестве LLM-модели
- Одинаковые наборы запросов для обеспечения сопоставимости тестирования
- Те же карты ключевых слов и карты знаний
- Суперкомпьютер НИУ ВШЭ для обучения и вывода модели с использованием его высокопроизводительной вычислительной (HPC) инфраструктуры
- Идентичный программный стек

Эксперимент с Transformers (baseline)

Method Name	Mean Time per 1 request(s)	Overall Time (s)
Transformers Experiment	141	707

Тип обработки: Синхронный, **Библиотека:** HuggingFace Transformers

- Каждый запрос обрабатывается последовательно, без параллелизма.
- Прямое взаимодействие с моделью через `pipeline` HuggingFace.
- Высокая задержка из-за блокирующей обработки.
- Низкая эффективность при высокой нагрузке.

Эксперимент с AsyncLLMEngine

Method Name	Mean Time per 1 request(s)	Overall Time (s)
Transformers Experiment	141	707
AsyncLLMEngine	72	365

Тип обработки: Асинхронный + динамический батчинг, **Библиотека:** vLLM

- Используется **AsyncLLMEngine** для параллельной обработки запросов.
- Оптимизированные CUDA-ядра для ускорения вычислений.
- Поддержка динамического батчинга (объединение запросов в группы).
- Ускорение в **2 раза** по сравнению с синхронным подходом.
- Лучшая утилизация GPU.

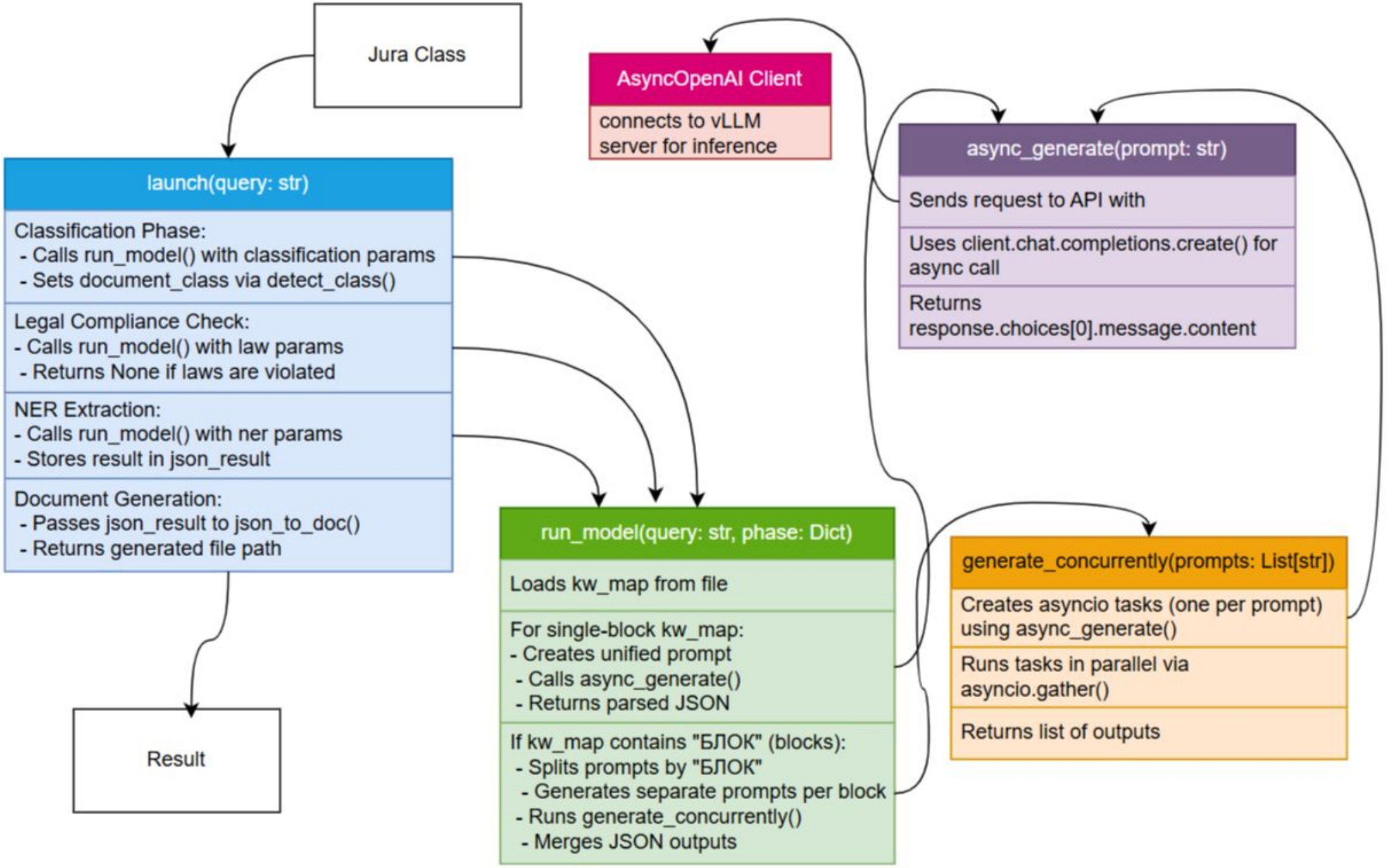
Эксперимент с AsyncOpenAI

Method Name	Mean Time per 1 request(s)	Overall Time (s)
Transformers Experiment	141	707
AsyncLLMEngine	72	365
AsyncOpenAI	8	40

Библиотека: AsyncOpenAI (клиент для vLLM API)

Тип обработки: Асинхронный + API-вызовы

- Взаимодействие с моделью через REST API (vLLM сервер).
- Полностью асинхронный клиент (`asyncio.gather` для параллельных запросов).
- Отдельный сервер для инференса (можно масштабировать).
- **Скорость в 17 раз выше**, чем у синхронного подхода.


Jura Class
AsyncOpenAI Client
connects to vLLM server for inference
launch(query: str)
Classification Phase:
- Calls run_model() with classification params
- Sets document_class via detect_class()
Legal Compliance Check:
- Calls run_model() with law params
- Returns None if laws are violated
NER Extraction:
- Calls run_model() with ner params
- Stores result in json_result
Document Generation:
- Passes json_result to json_to_doc()
- Returns generated file path
Result
run_model(query: str, phase: Dict)
Loads kw_map from file
For single-block kw_map:
- Creates unified prompt
- Calls async_generate()
- Returns parsed JSON
If kw_map contains "БЛОК" (blocks):
- Splits prompts by "БЛОК"
- Generates separate prompts per block
- Runs generate_concurrently()
- Merges JSON outputs
async_generate(prompt: str)
Sends request to API with
Uses client.chat.completions.create() for async call
Returns response.choices[0].message.content
generate_concurrently(prompts: List[str])
Creates asyncio tasks (one per prompt) using async_generate()
Runs tasks in parallel via asyncio.gather()
Returns list of outputs

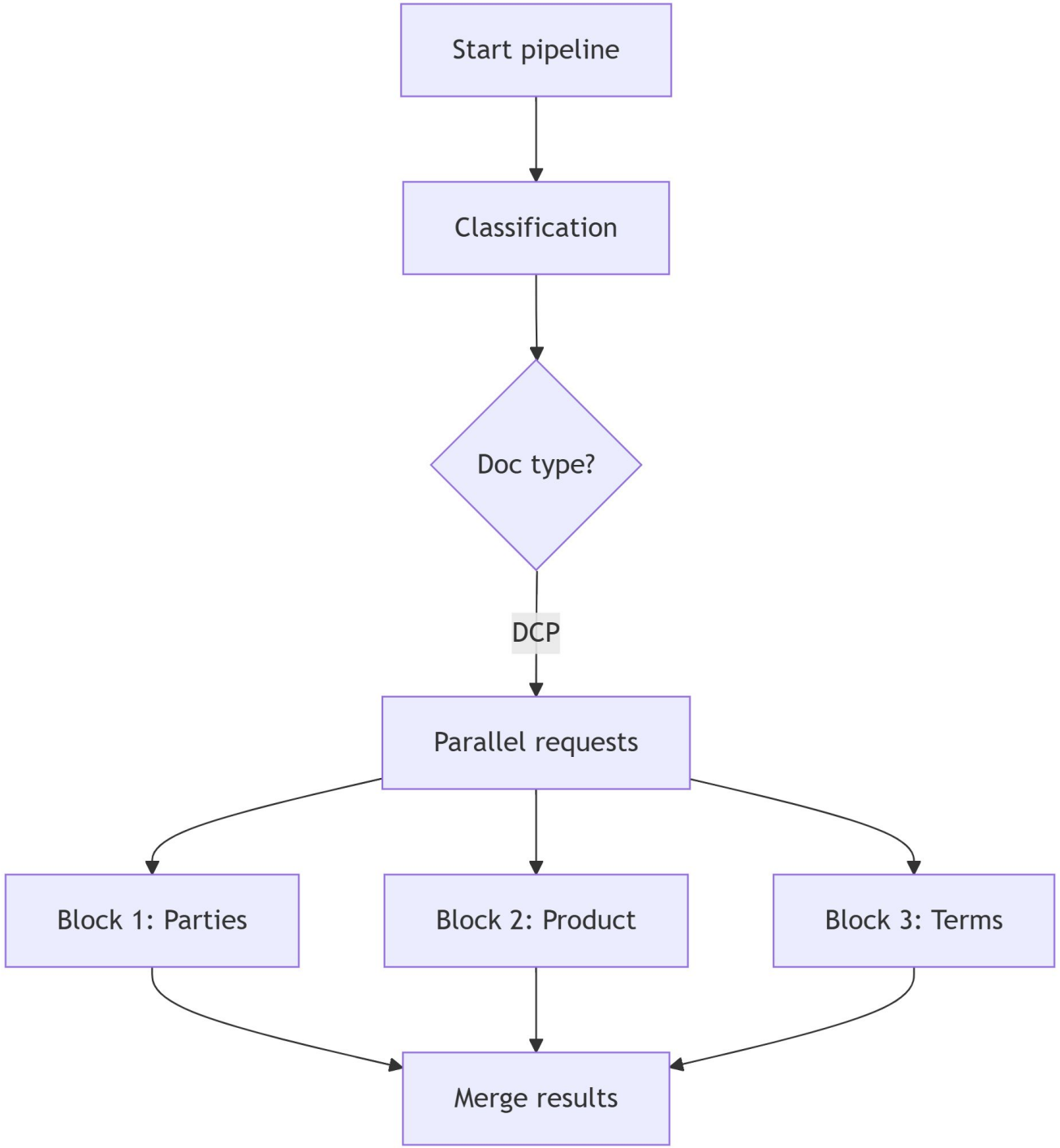
Эксперимент с AsyncOpenAI+StructuredOutput

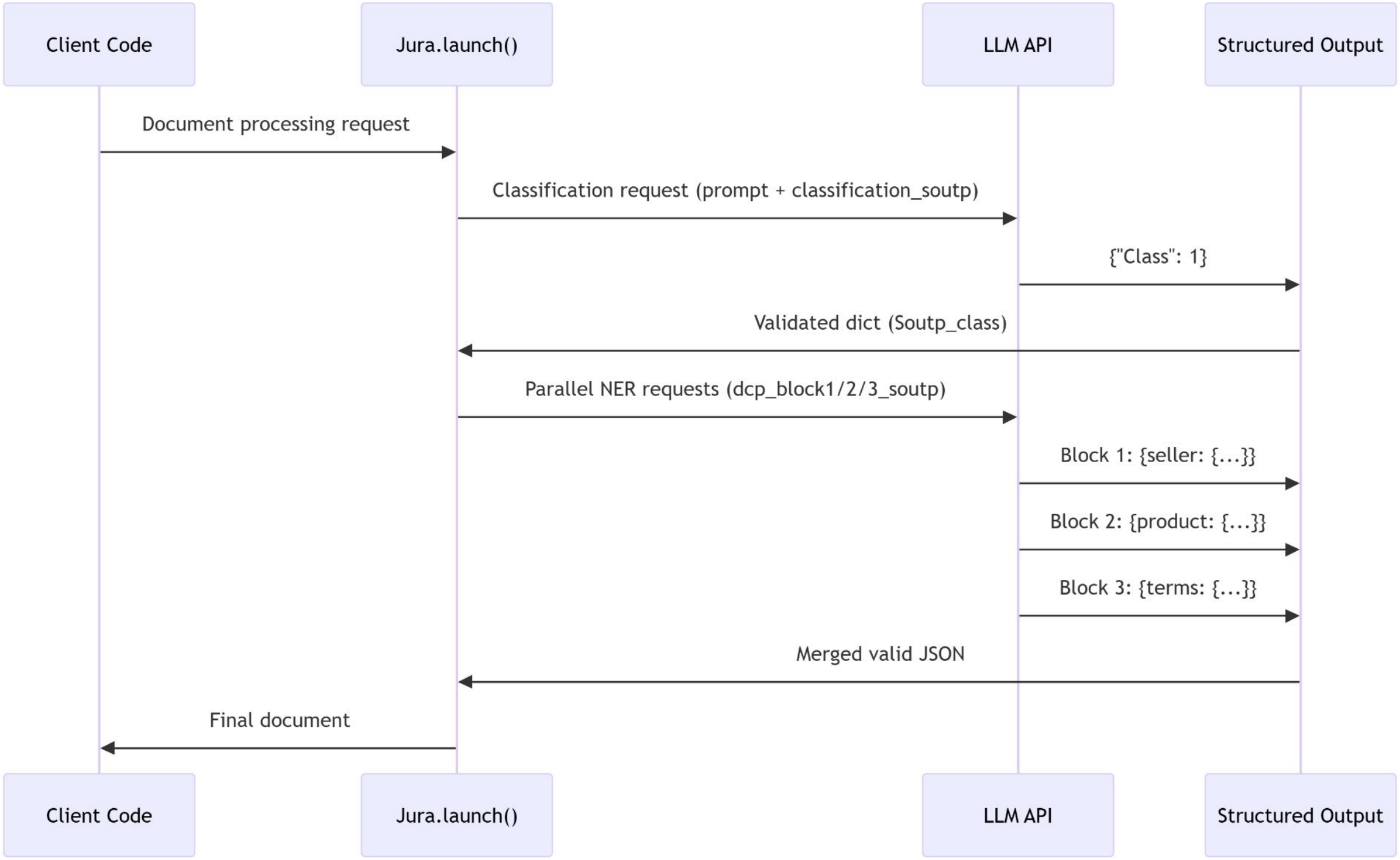
Method Name	Mean Time per 1 request(s)	Overall Time (s)
Transformers Experiment	141	707
AsyncLLMEngine	72	365
AsyncOpenAI	8	40

Библиотеки: AsyncOpenAI + Pydantic, **Тип обработки:** Асинхронный

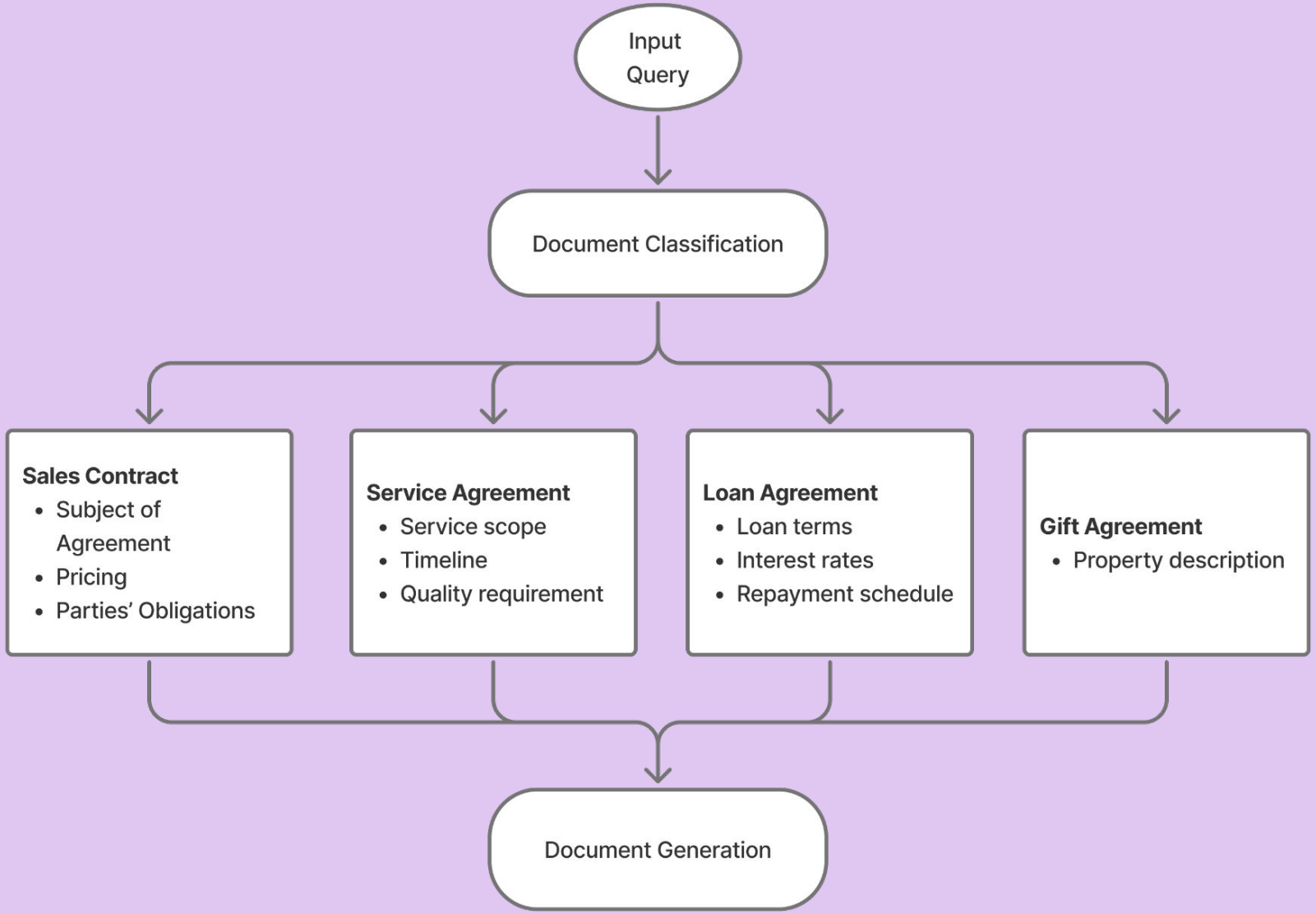
- Добавлена строгая валидация выходных данных через **Pydantic**.
- Автоматическое исправление JSON (если модель ошибается).
- Четкая схема выходных данных для юридических документов.

Method Name	Mean Time per 1 request(s)	Overall Time (s)
AsyncOpenAI	15	1575
Structured Output+AsyncOpenAI	18	1883


Start pipeline
Classification
Doc type?
DCP
Parallel requests
Block 1: Parties
Block 2: Product
Block 3: Terms
Merge results


Client Code
Jura.launch()
LLM API
Structured Output
Document processing request
Classification request (prompt + classification_soutp)
{"Class": 1}
Validated dict (Soutp_class)
Parallel NER requests (dcp_block1/2/3_soutp)
Block 1: {seller: {...}}
Block 2: {product: {...}}
Block 3: {terms: {...}}
Merged valid JSON
Final document
Client Code
Jura.launch()
LLM API
Structured Output

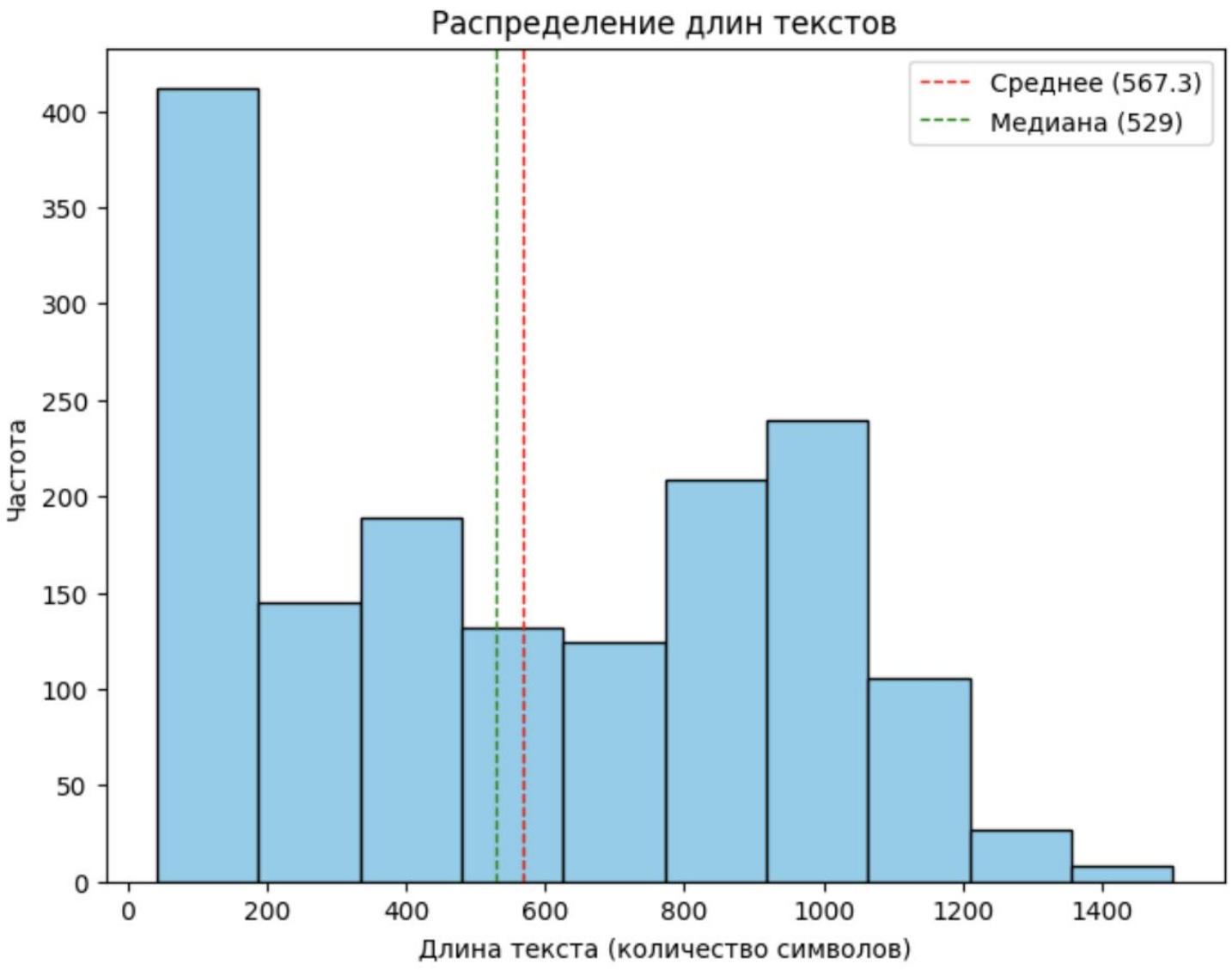
Улучшение классификации



Датасет

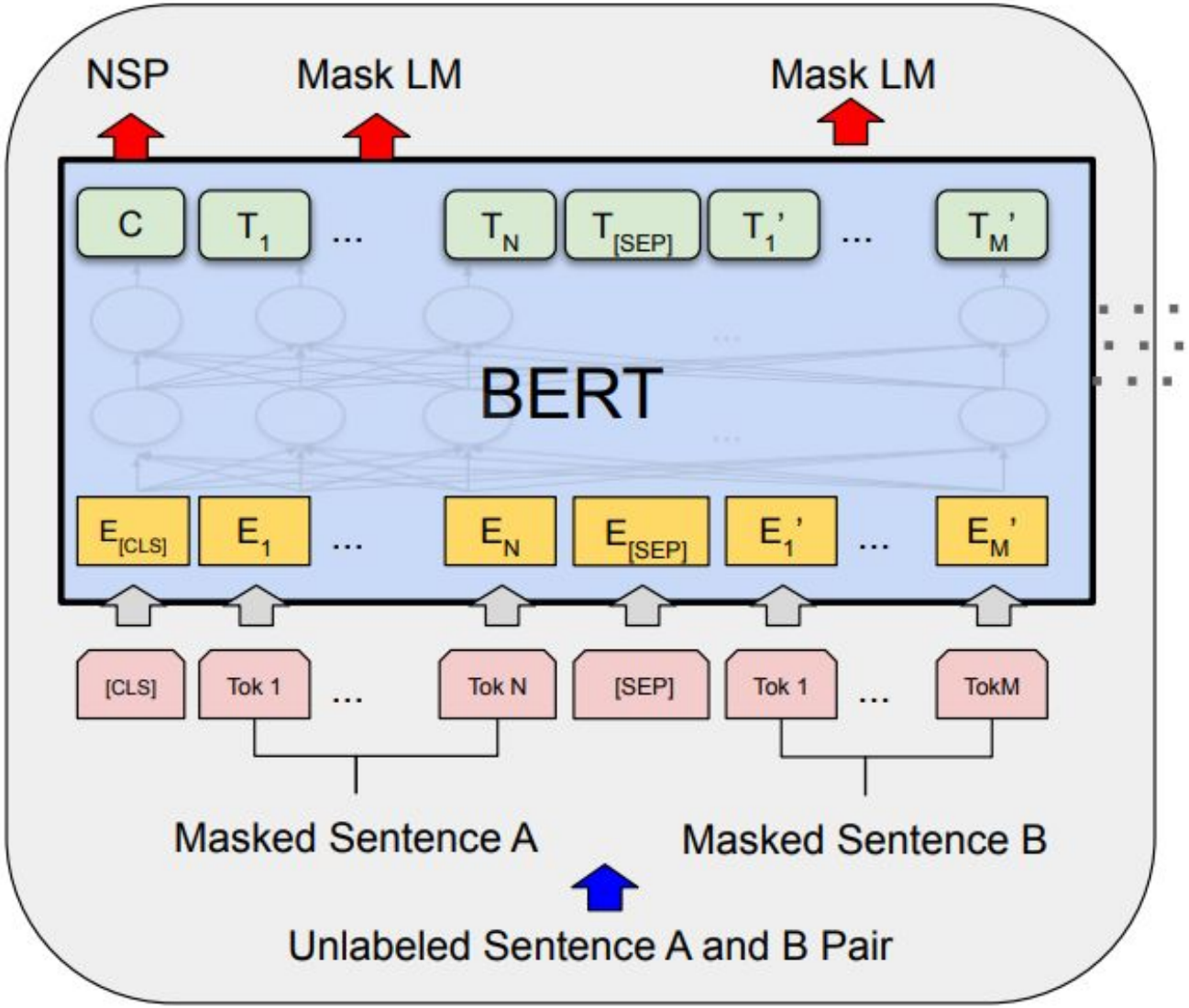
- Синтетический датасет, сгенерированный путем использования опенсорсных LLM-моделей

- Train: 1591 строка
  Test: 240 строк

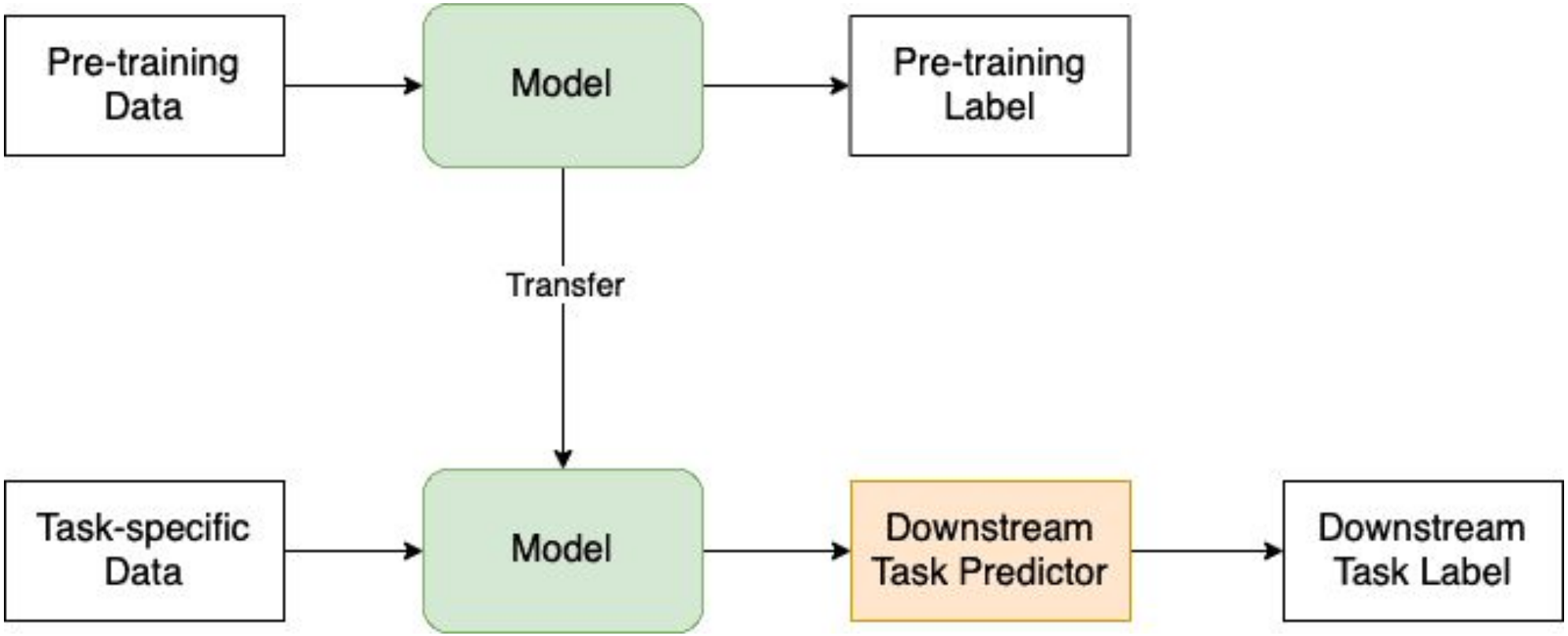


Выбор модели

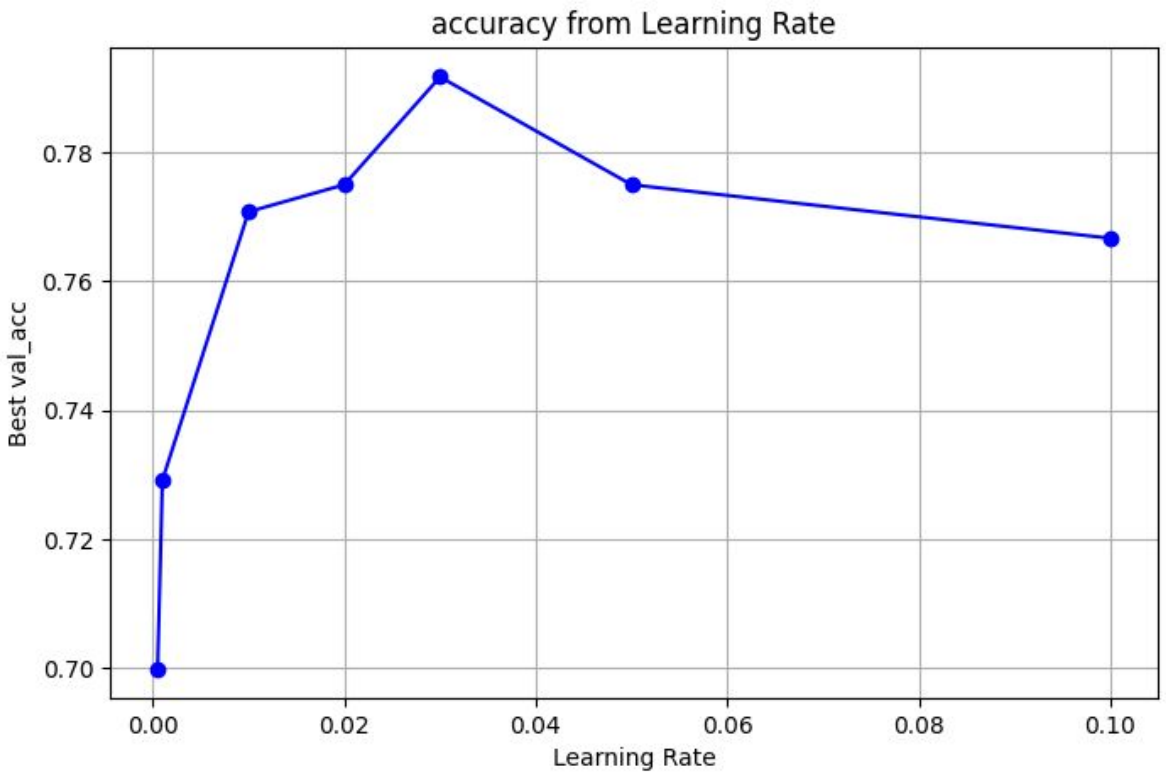
- Поддержка русского языка
- Готовые решения для дообучения и классификации
- Много предобученных моделей в открытом доступе
- BERT основан на современных трансформерах
- BERT для понимания текста, GPT — для генерации

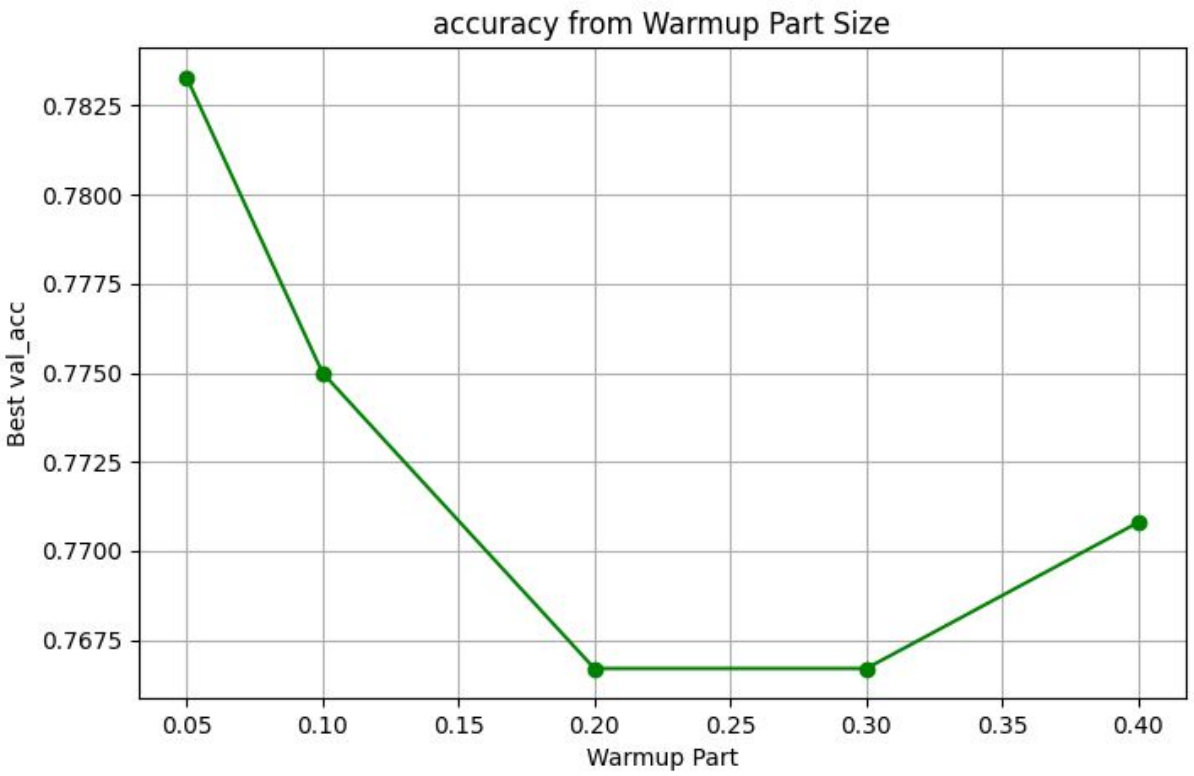


Transfer Learning

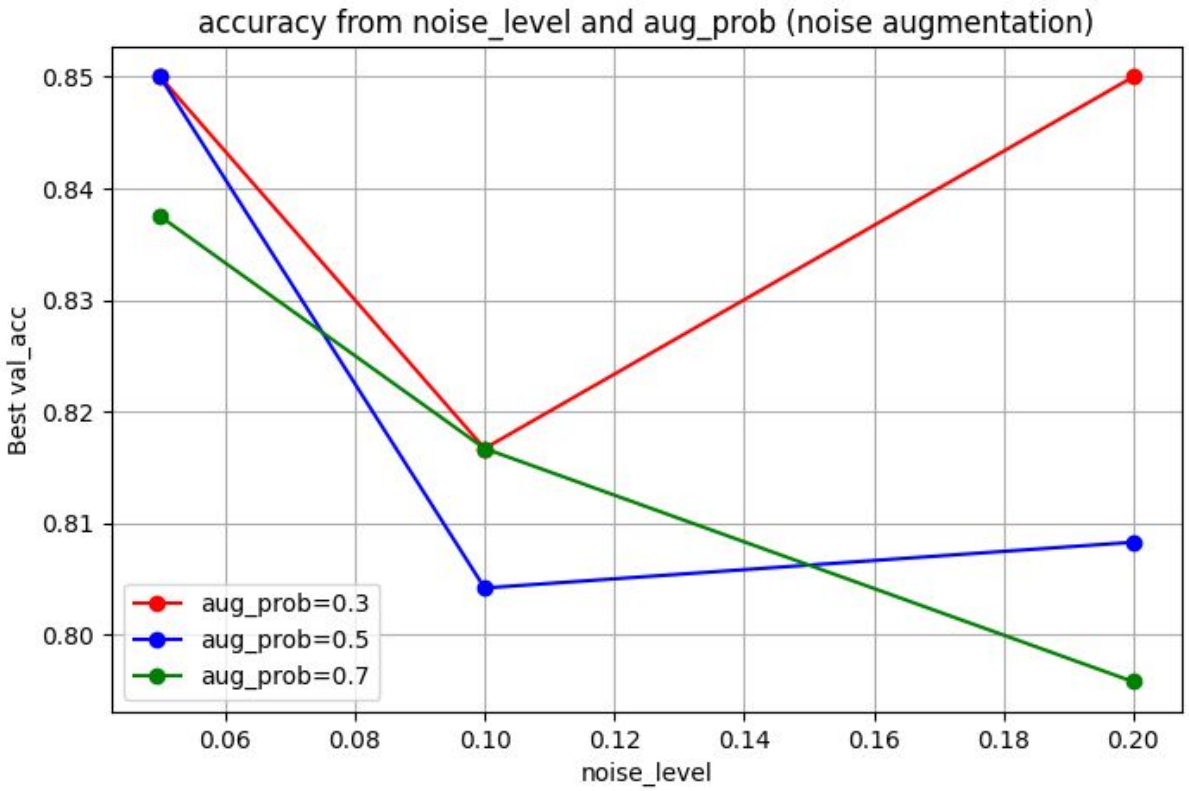


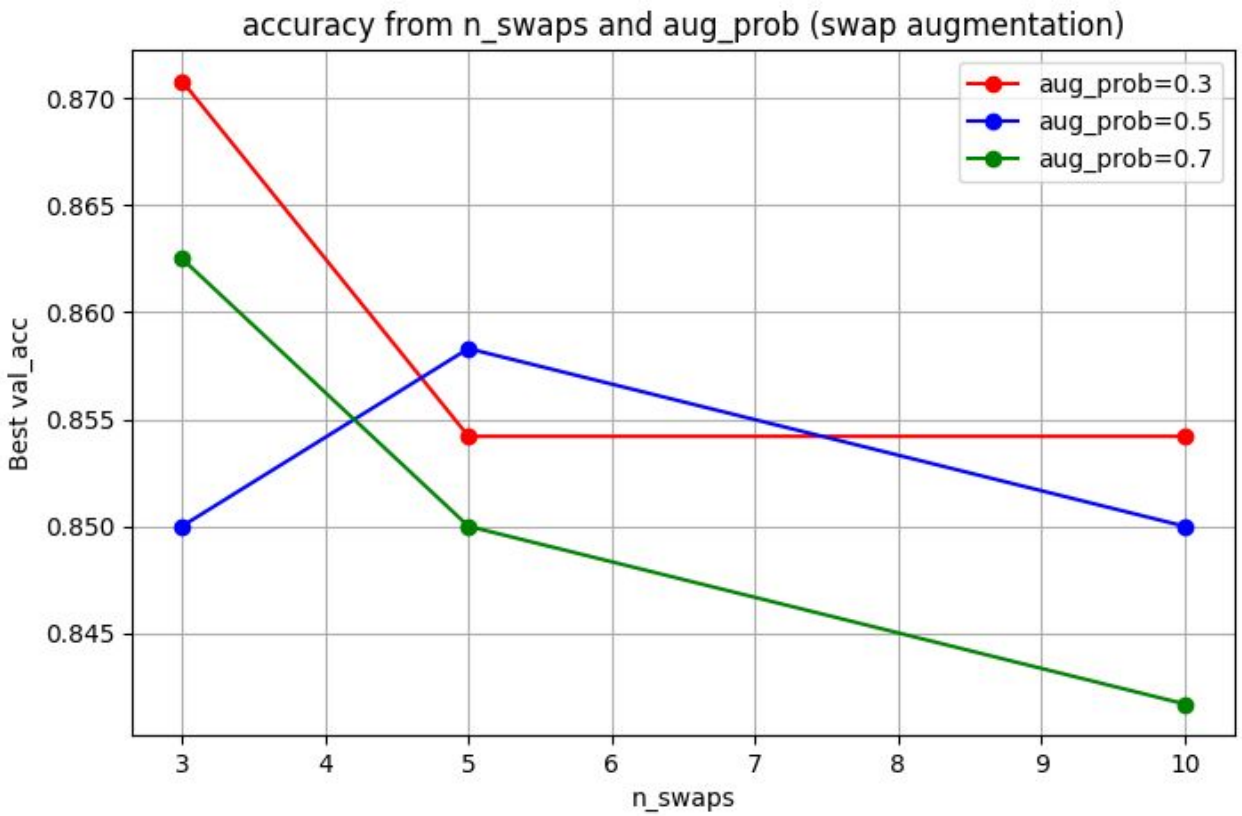
Эксперименты



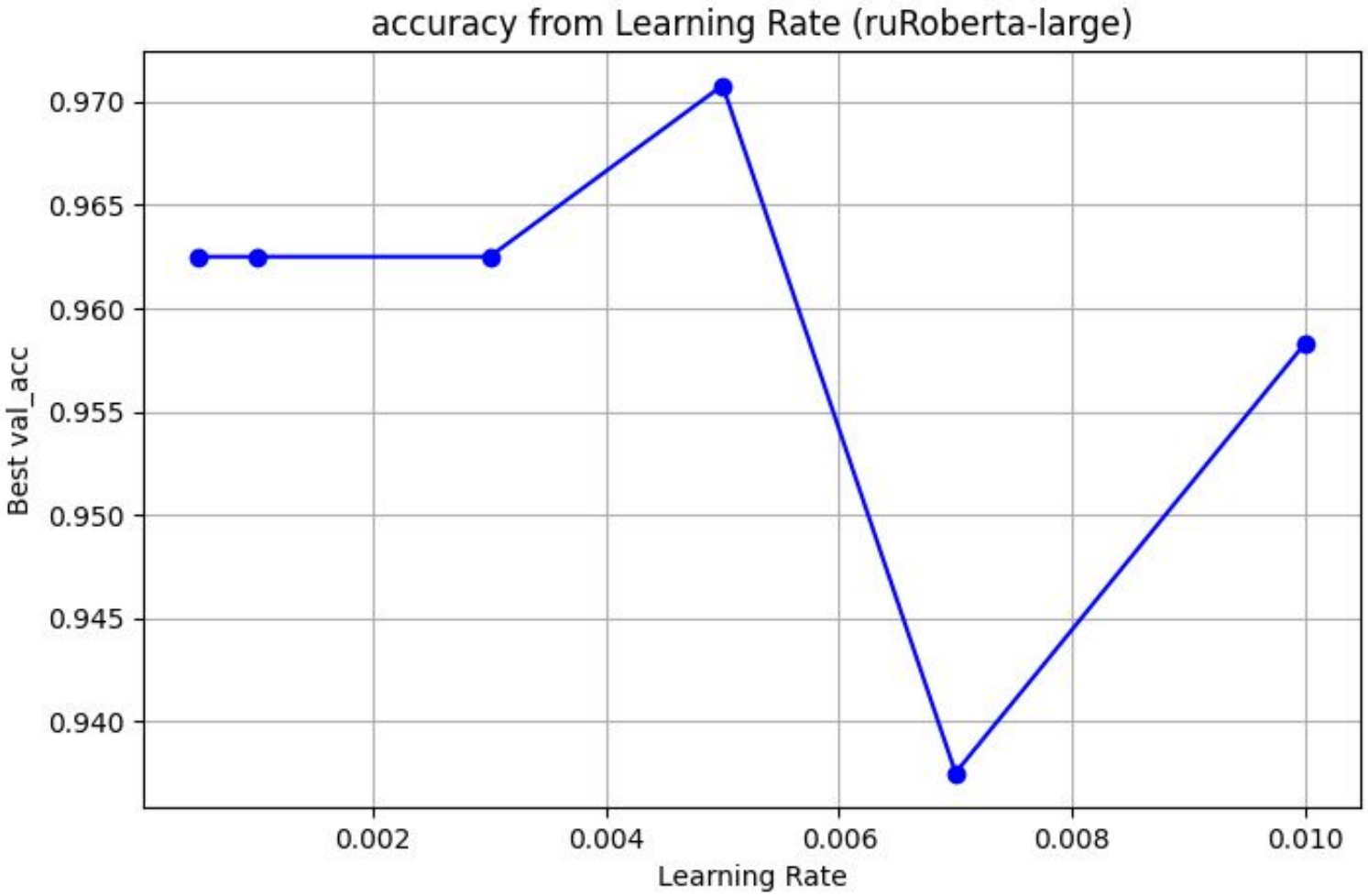


Эксперименты — Аугментации





Эксперименты — ruRoberta-large



Заключение

1) **Архитектурные улучшения**

2) **Оптимизация производительности**

3) **Технические изменения: асинхронность и параллельность**

4) **Повышение точности классификации**

Направления работы

- Parameter-Efficient Fine-Tuning
- Продолжение постановки экспериментов с классификатором договоров
- Интеграция с российскими LLM-моделями
- Расширение ряда доступных для генерации договоров
- Многоязычная поддержка

OKH