

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное  
образовательное учреждение высшего образования  
«Самарский национальный исследовательский университет  
имени академика С.П. Королева»  
(Самарский университет)

Институт информатики и кибернетики  
Кафедра технической кибернетики

ЛАБОРАТОРНАЯ РАБОТА № 2  
Применение `p8n` для реализации пайплайна обработки видео с  
использованием LLM моделей.

по курсу  
Инженерия данных

Группа 6232

Студент \_\_\_\_\_ Е.В. Заров  
(подпись)

Преподаватель \_\_\_\_\_ Р.А. Парингер  
(подпись)

## АРХИТЕКТУРА

N8n выполняет роль центрального управляющего компонента. Через него проходит весь процесс: получение видео из Telegram, извлечение аудио, распознавание речи, перевод субтитров и формирование итогового файла. Состояние и данные для работы n8n хранятся в PostgreSQL, а внешний доступ к системе обеспечивается через Traefik, который выполняет роль обратного прокси и отвечает за шифрование соединений.

После получения сообщения в Telegram workflow определяет, пришёл ли файл напрямую или была передана ссылка. В зависимости от этого выбирается способ скачивания видео. Затем файл сохраняется в локальную папку, и с помощью ffmpeg из него извлекается аудио. Полученное аудио отправляется во внешний STT-сервис на основе FastAPI и faster-whisper, который возвращает английские субтитры в формате SRT.

Далее субтитры передаются в цепочку LangChain, которая через OpenRouter отправляет текст на перевод в русскую версию, сохраняя структуру SRT. Готовый файл сохраняется, после чего снова используется ffmpeg для встраивания русских субтитров в видео. Итоговый ролик отправляется обратно пользователю в Telegram. Такая схема обеспечивает последовательную и устойчивую обработку видео от получения до финального результата.

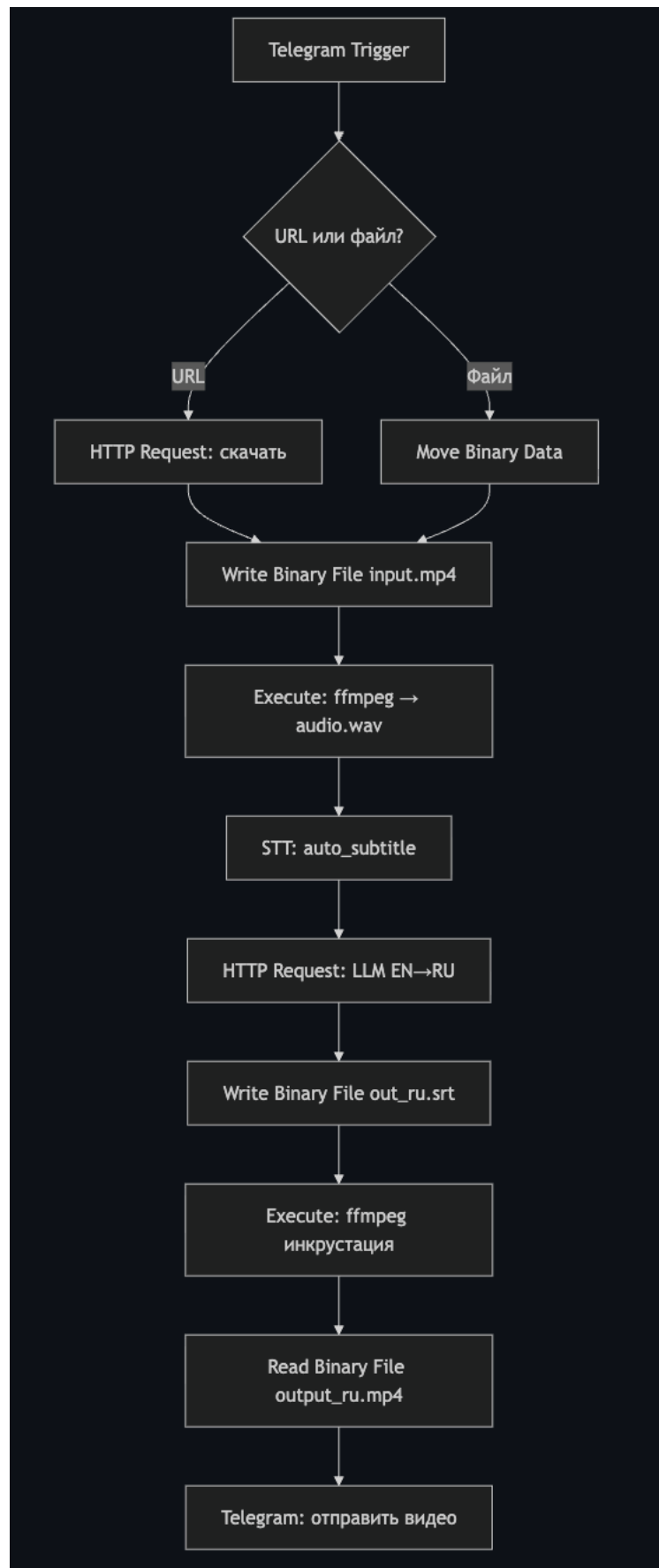


Рисунок 1 – Схема воркфлоу

## РЕЗУЛЬТАТЫ

На рисунке 2 представлен скриншот реализованного воркфлоу из n8n.

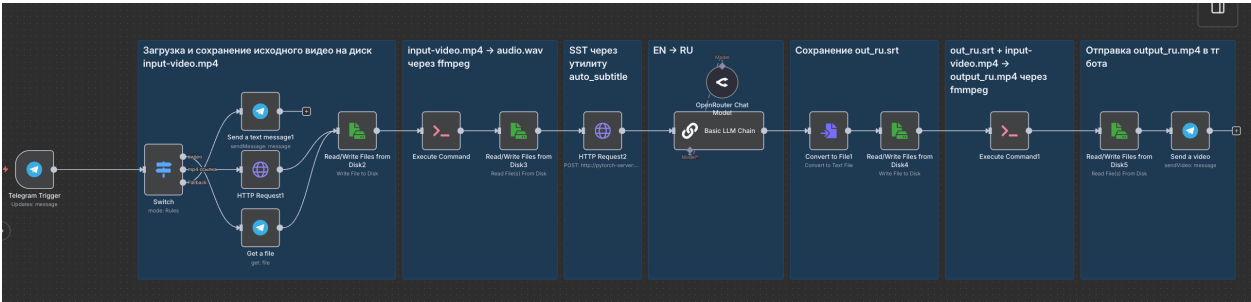


Рисунок 2 – Воркфлоу из n8n

На рисунке 3 показан пример работы Telegram-бота. Пользователь отправляет видео с английской речью, а в ответ получает тот же ролик, но уже с добавленными русскими субтитрами.

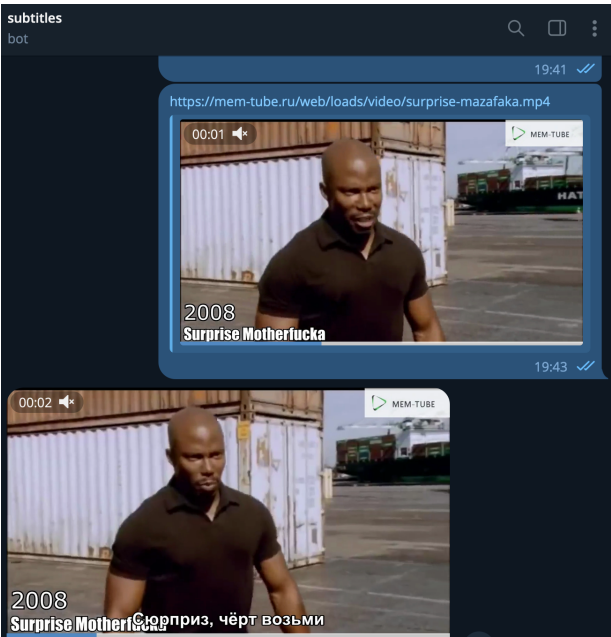


Рисунок 3 – Пример работы tg бота

На рисунке 4 представлены запущенные контейнеры, необходимые для корректной работы приложения.

```
root@cfaringdev:/tmp/ssau-data-engineering-lab2-dev# docker compose ps
```

NAME	IMAGE	COMMAND	SERVICE	CREATED	STATUS	PORTS
ssau-data-engineering-lab2-dev-n8n-1	local/n8n-ffmpeg:latest	"tini -- /docker-ent..."	n8n	15 hours ago	Up 15 hours	5678/
ssau-data-engineering-lab2-dev-postgres-1	postgres:16	"docker-entrypoint.s..."	postgres	15 hours ago	Up 15 hours	5432/
ssau-data-engineering-lab2-dev-pytorch-server-1	local/pytorch-server:latest	"uvicorn app:app --h..."	pytorch-server	15 hours ago	Up 15 hours	0.0.0
ssau-data-engineering-lab2-dev-traefik-1	traefik:v3.1	"/entrypoint.sh --pr..."	traefik	15 hours ago	Up 15 hours	0.0.0

Рисунок 4 – Запущенные контейнеры

## ВЫВОДЫ

Поскольку Telegram бот реализован в n8n так, что он должен реагировать на вебхук, из-за этого оставался либо вариант с лонг-пуллингом, либо полноценное развёртывание за HTTPS. Я арендовал сервер, настроил DNS/SSL и привязал n8n к поддомену - эта часть заняла ощутимое время, потому что Traefik сначала не поднимал 80/443 из-за фаервола, и пришлось вручную пробрасывать порты и проверять правила. Дополнительно возникла проблема с тем, что контейнер pytorch-server не мог достучаться до huggingface.co, поэтому модель скачал локально. Разбор настроек Traefik для поддомена n8n оказался еще одной трудностью - пришлось играть с правилами и сертификатами, прежде чем интерфейс заработал стабильно.

В ходе работы узнал, что telegram не дает скачивать видеофайлы крупнее примерно 20-30 МБ, поэтому желательно подавать ролики, которые не превышают этот размер. При загрузке видео по URL такого ограничения нет.

В процессе я освоил работу с обработкой видео, аудио и субтитров, научился применять инструменты ffmpeg и сервисы для автоматической генерации субтитров. Я впервые познакомился с n8n и получил практический опыт построения потоков автоматизации, взаимодействия со сторонними API и интеграции нескольких сервисов в единый конвейер.