**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**(Университет ИТМО)**

Факультет **Инфокоммуникационных технологий**

Образовательная программа **Интеллектуальные системы в гуманитарной сфере**

Направление подготовки (специальность) **45.03.04 Языковые модели и искусственный интеллект**

**К У Р С О В О Й П Р О Е К Т**

**по дисциплине «Инфокоммуникационные системы и технологии»**

на тему:

# Разработка телеграмм бота для анализа Youtube видео при помощи искусственного интеллекта

Обучающаяся Лебедева Анастасия Сергеевна, группа К3161

**Работа сдана**

Дата 30.12.2024

Санкт-Петербург 2024

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ](#kix.9r0j9clk338p).........................................................................................................2

[1 Описание проекта](#kix.91yu3wf0n8cd)............................................................................................3

[2 Процесс работы над проектом](#kix.w2rauyc6qgt3)........................................................................5

[2.1 Этапы реализации](#kix.7f5u5xz022fp)...............................................................................5

[2.2 Распределение задач](#kix.ljho9uw8thvo)...........................................................................5

[3 Реализация интеграции YandexGPT](#kix.3iqm3z1jbpn9)...............................................................7

[3.1 Постановка задачи](#kix.d3e98ci71od9)..............................................................................7

[3.2 Этапы решения](#kix.3jc0xgi8mdzm)...................................................................................7

[3.3 Особенности реализации](https://docs.google.com/document/d/1byaapcbY9qxFe1jzimuHawiFbDQFqqyKIYbHo671PQY/edit?tab=t.0#bookmark=id.vkdirmnscf6e)...................................................................8

[3.4 Возникшие трудности и их преодоление](#kix.h1c8nqjt4oc4).......................................10

[4 Тестирование и отладка](#tzluo8lcprrx).................................................................................12

5 [Взаимодействие с командой и руководителем](#kix.3lumqlhyuafx)............................................14 [ЗАКЛЮЧЕНИЕ](#kix.yv2dxvgovyvd).................................................................................................15

[СПИСОК](#kix.29erj75eym10) ИСПОЛЬЗОВАННЫХ [ИСТОЧНИКОВ](#kix.29erj75eym10)........................................16

[ПРИЛОЖЕНИ](#kix.9cw7pggaq2xm)E.................................................................................................17

**ВВЕДЕНИЕ**

Актуальность обусловлена стремительным ростом объема видеоконтента на платформе YouTube и возрастающей потребностью пользователей в быстром получении основной информации из видеоматериалов. В современном мире, где время является критическим ресурсом, автоматизированный анализ видеоконтента с помощью искусственного интеллекта становится необходимым инструментом для эффективной работы с информацией.

Разработанный в рамках проекта телеграмм-бот предназначен для широкого круга пользователей: исследователей и аналитиков, студентов и преподавателей, контент-менеджеров и маркетологов, обычных пользователей, желающих быстро ознакомиться с содержанием видео.

Целью проекта является разработка телеграмм-бота для анализа YouTube видео с использованием искусственного интеллекта, способного предоставлять пользователям краткую сводку содержания видеоматериалов на выбранном языке.

Для достижения поставленной цели были определены следующие задачи:

1. Реализация доступа к YouTube видео через API

2. Разработка базового функционала телеграмм-бота

3. Интеграция языковой модели YandexGPT

4. Создание и внедрение алгоритма анализа видео

5. Разработка удобного пользовательского интерфейса

6. Тестирование и оптимизация работы системы

7. Развертывание бота на сервере

**1 Описание проекта**

Разработанный проект представляет собой телеграмм-бота для анализа видеоконтента с платформы YouTube. Система построена на основе микросервисной архитектуры и включает несколько взаимосвязанных компонентов: Telegram бот как интерфейс взаимодействия с пользователем, Flask веб-сервер для обработки запросов, модуль работы с YouTube Transcript API и модуль интеграции с YandexGPT API.

Взаимодействие между компонентами осуществляется посредством HTTP-запросов с использованием REST API. Основной функционал системы включает обработку пользовательских команд (/start, /help, /settings), автоматическое распознавание и валидацию YouTube-ссылок, получение и анализ транскрипта видео, а также поддержку русского и английского языков интерфейса.

Для реализации проекта использовался стек технологий на базе Python с применением фреймворка Flask для веб-сервера и библиотеки python-telegram-bot для создания бота.

Процесс работы системы построен следующим образом: пользователь отправляет ссылку на YouTube видео в чат с ботом, где происходит первичная валидация ссылки. При успешной валидации формируется запрос к веб-серверу, который получает транскрипт видео через YouTube Transcript API. Полученный текст обрабатывается с помощью YandexGPT API для создания краткой сводки содержания, которая затем возвращается пользователю. В случае возникновения ошибок на любом этапе пользователь получает соответствующее уведомление.

Особое внимание в проекте уделено обеспечению надежности и удобства использования. Система способна обрабатывать различные форматы YouTube-ссылок, предоставляет понятные сообщения об ошибках и имеет интуитивно понятный интерфейс. Многоязычная поддержка позволяет анализировать видео независимо от языка оригинала и получать результаты на предпочтительном для пользователя языке (русский и английский). Наглядная схема работы телеграмм бота представлена на рисунке 7.

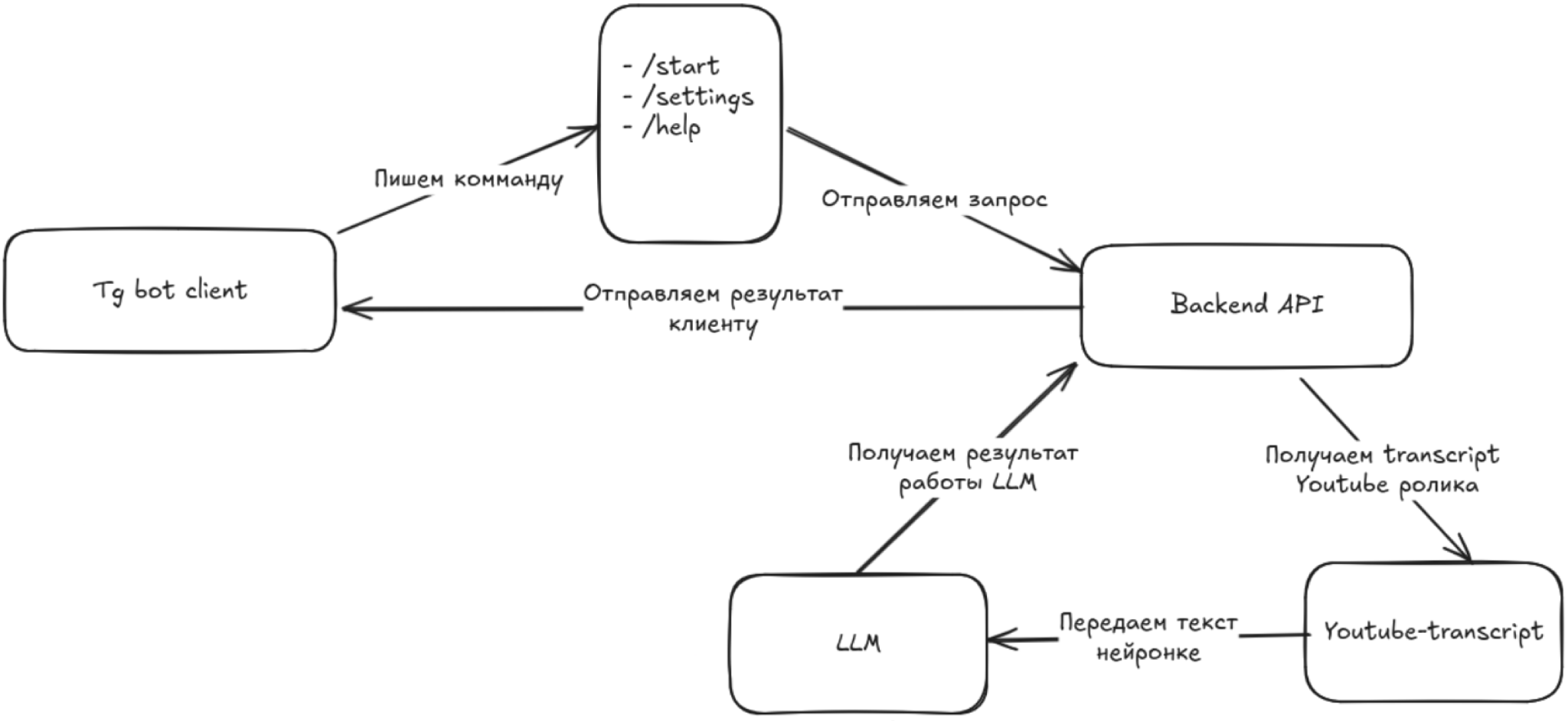


Рисунок 1 – Схема работы телеграмм бота

**2 Процесс работы над проектом**

**2.1 Этапы реализации**

Работа над проектом велась в период с ноября по декабрь 2024 года и была разделена на несколько последовательных этапов. На начальном этапе было получено необходимое оборудование и программное обеспечение, включая доступ к API сервисов. Затем была реализована основная функциональность по работе с YouTube видео и разработана базовая версия телеграмм-бота. После этого были созданы алгоритмы анализа видео с помощью YandexGPT и внедрены в систему. Заключительными этапами стали разработка пользовательского интерфейса и размещение бота на сервере.

**2.2 Распределение задач**

В рамках проекта задачи были распределены между участниками команды следующим образом:

- Первый участник отвечал за разработку базового функционала бота, включая обработку команд (/start, /help, /settings) и реализацию механизма распознавания YouTube-ссылок с использованием регулярных выражений.

- Второй участник занимался созданием модуля для обработки YouTube-ссылок с использованием python-transcript-api, реализуя асинхронную функцию для получения и форматирования текста видео.

- Третий участник разрабатывал веб-сервер на базе Flask с endpoint'ом /summarize для обработки YouTube-ссылок и получения текста видео.

- Моей задачей была интеграция YandexGPT в веб-сервер для анализа текста видео, включая разработку промпта и оптимизацию параметров запроса.

- Четвертый участник реализовывал взаимодействие между ботом и веб-сервером, включая отправку POST-запросов и обработку ответов.

**3 Реализация интеграции YandexGPT**

**3.1 Постановка задачи**

В рамках разработки системы для анализа YouTube видео передо мной была поставлена задача интеграции языковой модели YandexGPT в веб-сервер проекта. Основные требования к реализации включали:

- Добавление функционала запроса к LLM модели с текстом транскрипта видео

- Использование YandexGPT для анализа содержания

- Реализация промпта для получения краткой сводки видео

- Настройка параметра temperature=0 для улучшения качества ответов

- Использование библиотеки yandexgpt-python для взаимодействия с API

**3.2 Этапы решения**

Процесс реализации интеграции YandexGPT был разделен на следующие этапы:

Первый этап - подготовка и изучение инструментов:

- Изучение документации YandexGPT API

- Анализ возможностей библиотеки yandexgpt-python

- Тестирование базовых запросов к API

Второй этап - разработка базовой функциональности:

- Написание функции для взаимодействия с YandexGPT API

- Реализация промпта для анализа видео

- Настройка параметров запроса

Третий этап - интеграция с веб-сервером:

- Добавление обработчика запросов

- Реализация асинхронного выполнения запросов

- Обработка ошибок и исключительных ситуации

**3.3 Особенности реализации**

При реализации интеграции были учтены следующие особенности:

1. Формирование промпта

При разработке интеграции YandexGPT с веб-сервером особое внимание было уделено надежности и эффективности работы системы. В первую очередь был реализован механизм формирования запросов к API YandexGPT. Для этого использовалась библиотека yandexgpt-python, которая предоставляет удобный интерфейс для взаимодействия с API [1]. В рамках реализации был создан специальный промпт для анализа видео, который включает название видео и его текстовое содержание. Важной особенностью стала настройка параметра temperature=0, что позволило получать более последовательные и точные результаты анализа.

2. Настройка параметров запроса

Особое внимание было уделено оптимизации использования API. Для этого были настроены оптимальные параметры запросов, включая количество токенов и температуру генерации. Также была реализована система кэширования результатов для часто запрашиваемых видео, что позволило снизить нагрузку на API и ускорить работу системы в целом. [5]

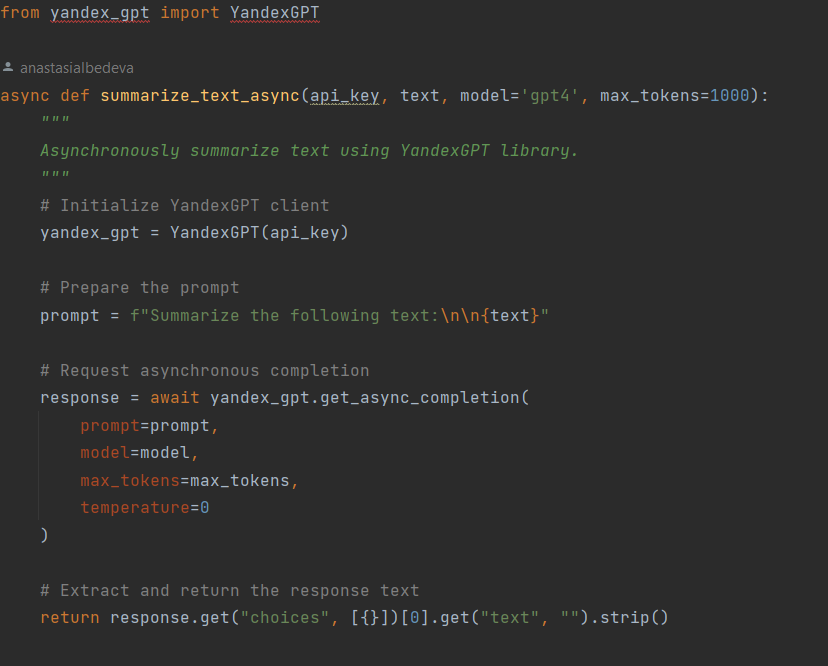


Рисунок 2 - Асинхронная функция с параметрами запроса

3. Обработка ответа

Для обеспечения надежности работы системы была реализована комплексная система обработки ошибок и автоматических повторных попыток. При возникновении сбоев в работе API система автоматически выполняет повторные попытки с экспоненциальной задержкой между ними, что позволяет справляться с временными проблемами связи или перегрузкой сервера [2]. Каждая попытка регистрируется в системе логирования для последующего анализа и отладки.

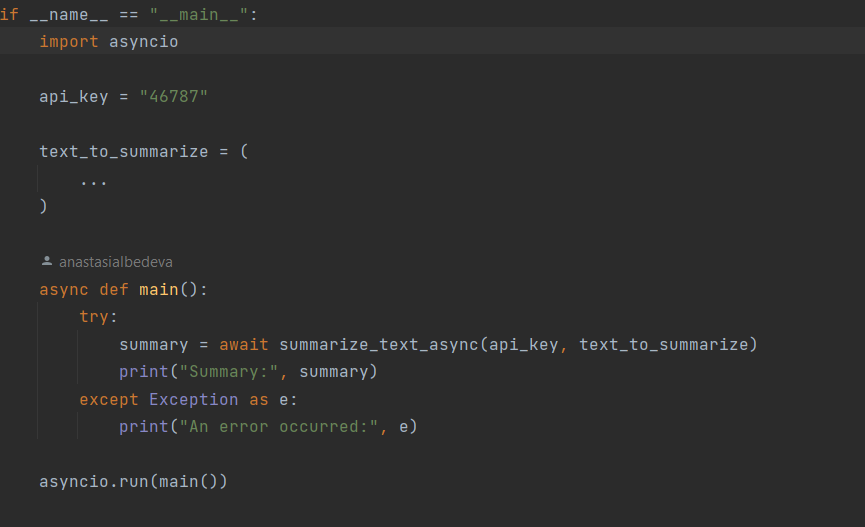


Рисунок 3 - Обработка ответа YandexGPT

**3.4 Возникшие трудности и их преодоление**

В процессе работы над интеграцией YandexGPT возникли следующие трудности:

1. Проблема с обработкой длинных текстов:

- Причина: ограничение на максимальную длину входного текста в API

- Решение: реализация разделения длинных текстов на части и их последовательная обработка [6]

2. Оптимизация качества ответов:

- Причина: начальные результаты анализа были недостаточно информативными

- Решение: настройка параметра temperature=0 и оптимизация структуры промпта

3. Асинхронная обработка запросов:

- Причина: необходимость обработки нескольких запросов одновременно

- Решение: использование асинхронных функций Python [2]

4. Обработка ошибок API:

- Причина: нестабильные ответы API при большой нагрузке

- Решение: реализация механизма повторных попыток и обработки исключений [4]

За время выполнения курсового проекта были получены ценные навыки и опыт:

- Работа с современными API и языковыми моделями

- Разработка асинхронных веб-приложений на Python

- Использование системы контроля версий Git в командной работе

- Навыки технической коммуникации и документирования кода

- Опыт интеграции различных сервисов в единую систему

**4 Тестирование и отладка**

Тестирование разработанной системы проводилось в несколько этапов с использованием различных подходов. На первом этапе выполнялось модульное тестирование отдельных компонентов системы. Для тестирования функционала взаимодействия с YandexGPT API использовались различные варианты входных данных, включая граничные случаи и потенциально проблемные ситуации.

В рамках интеграционного тестирования проверялось взаимодействие между компонентами системы:

- Корректность передачи данных между ботом и веб-сервером

- Правильность формирования и обработки запросов к YandexGPT

- Обработка ошибок и исключительных ситуаций

- Взаимодействие с внешними API (YouTube, Telegram)

В процессе тестирования были проверены следующие аспекты системы: Тестирование API интеграции:

- Корректность формирования промпта для YandexGPT

- Обработка ответов различной длины

- Поведение системы при недоступности API

- Качество получаемых результатов анализа

Проверка обработки ошибок:

- Реакция на некорректные URL

- Обработка таймаутов запросов

- Поведение при превышении лимитов API

- Восстановление после сбоев

По результатам тестирования были выявлены и устранены следующие проблемы:

- Оптимизирована обработка длинных текстов

- Улучшена обработка ошибок API

- Добавлена дополнительная валидация входных данных

- Реализован механизм повторных попыток при сбоях

Все обнаруженные проблемы были успешно решены, и система демонстрирует стабильную работу при различных сценариях использования.

**5 Взаимодействие с командой и руководителем**

Взаимодействие с командой строилось на принципах открытой коммуникации и взаимопомощи. Для координации работы использовались еженедельные онлайн-встречи для обсуждения прогресса, групповой чат в Telegram для оперативной коммуникации, GitHub для совместной работы над кодом, регулярные code review для обеспечения качества кода.

Особенно эффективным оказалось разделение задач с учетом сильных сторон каждого участника и регулярный обмен опытом между членами команды. При возникновении сложностей команда совместно находила решения, что способствовало не только выполнению проекта, но и профессиональному росту каждого участника.

Взаимодействие с руководителем проекта происходило на регулярной основе через:

- Еженедельные консультации по техническим вопросам

- Проверку промежуточных результатов

- Обсуждение возникающих проблем и путей их решения

- Корректировку направления работы при необходимости

Руководитель проекта продемонстрировал высокий профессионализм и компетентность в управлении проектом. Особенно стоит отметить четкую постановку задач и определение требований, конструктивную обратную связь по результатам работы, готовность помочь в решении технических вопросов , гибкий подход к организации рабочего процесса, поддержание мотивации команды на протяжении всего проекта.

Благодаря эффективному руководству и поддержке, проект был успешно завершен в установленные сроки, а все участники получили ценный опыт командной разработки программного обеспечения.

**ЗАКЛЮЧЕНИЕ**

В результате выполнения курсового проекта была успешно достигнута основная цель - разработан телеграмм-бот для анализа YouTube видео с использованием искусственного интеллекта. Все задачи, поставленные в техническом задании, были выполнены в полном объеме в установленные сроки. Созданный бот позволяет пользователям получать краткие сводки содержания видео на выбранном языке (русском или английском). В ходе работы над проектом были реализованы все основные компоненты системы: создан бот с базовыми командами и обработкой ссылок, разработан веб-сервер для взаимодействия с API, реализовано получение транскриптов видео, внедрен анализ содержания с помощью YandexGPT.

Мой личный вклад в проект заключался в реализации интеграции YandexGPT API в веб-сервер. В рамках этой задачи было реализовано взаимодействие с API, создан оптимальный промпт для анализа видео, настроены параметры для улучшения качества ответов и реализована обработка ошибок. Все основные требования технического задания были выполнены, система успешно прошла тестирование и готова к использованию. Разработанный бот обеспечивает удобный интерфейс взаимодействия через Telegram, поддерживает мультиязычность и предоставляет пользователям возможность быстрого получения основной информации из видеоматериалов.

**СПИСОК ИСТОЧНИКОВ**

1 Библиотека yandexgpt-python [Электронный ресурс]. URL: https://github.com/allseeteam/yandexgpt-python (дата обращения: 12.11.24).

2 Документация Python asyncio [Электронный ресурс]. URL: https://docs.python.org/3/library/asyncio.html (дата обращения: 24..11.24).

3 Flask Documentation [Электронный ресурс]. URL: https://flask.palletsprojects.com/ (дата обращения: 20.11.24).

4 Telegram Bot API [Электронный ресурс]. URL: <https://core.telegram.org/bots/api> (дата обращения: 15.11.24).

5 Yandex Cloud [Электронный ресурс].- URL: https://cloud.yandex.ru/docs/yandexgpt/ (дата обращения: 30.11.24).

6 YouTube Transcript API [Электронный ресурс]. URL: https://pypi.org/project/youtube-transcript-api/ (дата обращения: 05.12.24).

**ПРИЛОЖЕНИЕ**

**Техническое задание**

Сроки выполнения:

11.2024 (начало: месяц, год) – 12.2024 (окончание: месяц, год)

Цель проекта:

Разработать телеграмм-бота для анализа YouTube видео с использованием искусственного интеллекта.

Ключевые слова и словосочетания:

- Анализ видео YouTube

- API взаимодействия с нейросетью

- Искусственный интеллект

- Сводка содержания видео

- Таймкоды ключевых моментов видео

- Мультиязыковая поддержка (русский и английский языки)

- Телеграмм-бот (конечный продукт)

Требования к проекту: Бот должен принимать ссылки на ютуб видео и обрабатывать их. после обработки, юзер при помощи команд должен взаимодействовать с ботом и получать краткую информацию из видео. Также пользователь моет прочитать инструкцию по использованию бота.

Бот должен работать во всех официальных приложениях Telegram (мобильных, веб- и десктопных версиях).

Бот должен проверять валидность ссылок и выдавать сообщение об ошибке, если ссылка недействительна.

Пользовательский интерфейс бота должен быть интуитивно понятным.

Содержание работы (этапы по срокам):

| Этап работы | Срок выполнения | Документ о выполнении этапа |
| --- | --- | --- |
| Получить необходимое оборудование и ПО, API | 05.11.2024 | Выписка о получении API |
| Реализовать доступ к YouTube видео | 10.11.2024 | График подключения к youtube.com |
| Разработать оболочку бота (Python, YandexGPT) | 15.11.2024 | Ссылка на бота |
| Написать алгоритм анализа видео | 30.11.2024 | Отчет о работе алгоритма |
| Внедрить алгоритм анализа в бота | 02.12.2024 | Ссылка на бота |
| Создать удобный пользовательский интерфейс | 10.12.2024 | Ссылка на бота |
| Разместить бота на сервере | 12.12.2024 | Ссылка на бота |

Основной результат работы и форма его предоставления:

Результат: Телеграмм-бот для анализа YouTube видео, предоставляющий краткую сводку материала на языке запроса

Форма представления: Готовый к использованию бот, доступный в Telegram.