**НИТУ «МИСиС»**

**Кафедра инженерной кибернетики**

**ОТЧЕТ**

по

**ЛАБОРАТОРНОЙ РАБОТЕ №1**

*«Разработка демонстрационного прототипа приложения для решения специализированной задачи интеллектуальной обработки и анализа информации с использованием современных ИИ-сервисов»*

учебная дисциплина «Методы искусственного интеллекта»

**Группа: БПМ**-**21-1**

**Учащийся:** **Ишмухамедов А.А.**

**Преподаватель: Хонер П.Д.**

**Оценка:**

**Дата защиты:**

**2024 г.**

Оглавление

[ВВЕДЕНИЕ 3](#_Toc184780678)

[1. ОПИСАНИЕ ЗАДАЧИ 3](#_Toc184780679)

[1.1. Основное содержание задачи 3](#_Toc184780680)

[1.2. Решение включает: 3](#_Toc184780681)

[1.3. Особенности решения 4](#_Toc184780682)

[3. ВЫБРАННЫЕ СРЕДСТВА ДЛЯ РАЗРАБОТКИ 5](#_Toc184780683)

[4. ИСТОЧНИКИ ИСХОДНЫХ ДАННЫХ 6](#_Toc184780684)

[5. РЕЗУЛЬТАТЫ 6](#_Toc184780685)

[6. ВЫВОДЫ 7](#_Toc184780686)

[ИСТОЧНИКИ 8](#_Toc184780687)

ВВЕДЕНИЕ

Целью данной лабораторной работы является выработка устойчивых навыков взаимодействия с современными ИИ-сервисами на уровне программного кода для решения задачи интеллектуальной обработки информации, а также оценка качества работы таких сервисов на примере преобразования аудиофайлов в текст.

В рамках данной работы рассматривалась задача преобразования аудиофайлов в текст с использованием современных ИИ-сервисов. Исследуемые API — Whisper и AssemblyAI. Проблемная область относится к обработке текстов на естественном языке (NLP). Цель — определить лучший сервис по качеству распознавания.

1. ОПИСАНИЕ ЗАДАЧИ
   1. Основное содержание задачи

Изучение возможности преобразования аудиофайлов в текст с использованием современных ИИ-сервисов для обработки естественного языка (NLP). Рассматриваемая задача относится к области интеллектуальной обработки информации.

Задача заключается в преобразовании аудиофайлов, содержащих речевую и музыкальную информацию (голос в сопровождении музыки), в текстовые расшифровки.

* 1. **Решение включает**:

1. Подключение к API ИИ-сервисов (OpenAI Whisper и AssemblyAI).
2. Отправку аудиофайлов для обработки.
3. Получение текстовых результатов.
4. Сравнение качества работы двух сервисов на основании точности расшифровки.
5. Исходные данные для лабораторной работы включают аудиофайлы, созданные лично, а также музыкальные композиции и видеоматериалы, взятые из открытых источников.
6. Для эксперимента использовались 10 аудиофайлов в формате MP3 с разнообразным содержанием.

Примеры содержаний аудиофайлов:

* Короткая речь (мужской и женский голос).
* Речь с фоновым шумом.
* Детский голос с дефектами речи.
* Музыкальные композиции (рэп, поп).
* Синхронно произнесенные фразы мужским и женским голосом.

Примеры результатов

Результатом обработки каждого аудиофайла является текстовая расшифровка. Например:

* «Ты тестируешь приложение для преобразования голоса в текст. Просто загрузи любой mp3 файл и выбери один из двух API, чтобы увидеть результат. Это поможет тебе сравнить их работы и выбрать лучший вариант.
* «Мы с тобой шёпотом, шёпотом Спрашивали, что потом, что потом будет Шёпотом, шёпотом Не хочу кричать о том, что друг друга забудем»
  1. Особенности решения

Использованы два API:

1. OpenAi Whisper API, интегрированный через RapidAPI [1]
2. AssemblyAI API, использована библиотека assemblyai для языка программирования Python [2]

Программа предоставляет удобный и простой интерфейс на Python с использованием библиотеки PyQt5. Пользователь может выбирать API, загружать файлы и получать результаты в графическом интерфейсе.

Реализован механизм обработки ошибок и визуализация результата.

Таким образом, задача позволяет оценить качество работы современных ИИ-сервисов для преобразования аудио в текст, а также выявить особенности их применения в различных условиях.

1. ВЫБРАННЫЕ СРЕДСТВА ДЛЯ РАЗРАБОТКИ

Для разработки программного обеспечения были использованы следующие инструменты и технологии:

1. Whisper API — современный программный интерфейс (API), предоставляющий функционал преобразования аудиофайлов в текст. API интегрировался через платформу RapidAPI с использованием стандартных методов HTTP-запросов.
2. AssemblyAI API — аналогичный сервис, обеспечивающий преобразование аудио в текст, однако его использование через платформу RapidAPI вызвало ошибку HTTP 500 (рис. 1). Для работы с данным API применялась официальная библиотека assemblyai [2] для языка программирования Python, что позволило организовать прямое взаимодействие с сервисом.

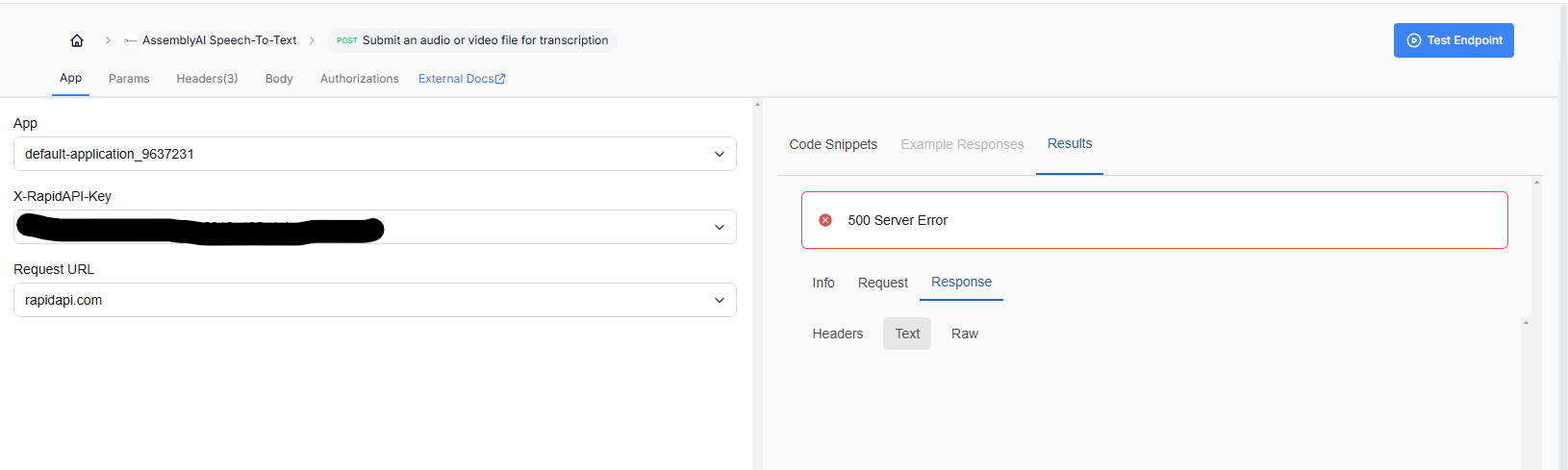


Рис. 1. Ошибка на стороне сервера при подключении к AssemblyAI API через RapidApi

1. Язык программирования Python — основной инструмент разработки, обеспечивающий гибкость интеграции и обработки данных.
2. Среда разработки: Visual Studio Code — кроссплатформенная интегрированная среда разработки, использованная для написания, отладки и тестирования программного кода.
3. Библиотеки:

* requests — для выполнения HTTP-запросов к API.[3]
* dotenv — для безопасного хранения конфиденциальной информации, включая API-ключи, в файле .env.
* PyQt5 — библиотека для создания графического пользовательского интерфейса приложения, предоставляющего пользователю возможность выбора API, загрузки файлов и отображения результатов. [4]

Использование данных инструментов позволило реализовать программное обеспечение, соответствующее требованиям поставленной задачи, а также обеспечить удобство работы пользователя в процессе тестирования.

1. ИСТОЧНИКИ ИСХОДНЫХ ДАННЫХ

Для тестирования были взяты аудиозаписи из свободных источников и записаны собственные аудиофайлы с заранее подготовленным текстом. В итоге среди исходных данных 10 аудиозаписей в формате mp3:

1. Мужской голос, быстрая речь, длительность
2. Мужской голос, медленная речь, длительность
3. Женский голос, обычная речь, длительность
4. Женский голос, быстрая речь, длительность
5. Мужской и женский голос, синхронно, длительность

Общий текст для 5 вышеописанных аудиозаписей: “Привет! Ты тестируешь приложение для преобразования голоса в текст. Просто загрузи любой mp3 файл и выбери один из двух API, чтобы увидеть результат. Это поможет тебе сравнить их работу и выбрать лучший вариант.”

1. Женский голос, быстрая речь, длительность
2. Отрывок из песни Сергея Лазарева “Шепотом”, длительность

Текст отрывка:

«Мы с тобой шёпотом, шёпотом

Спрашивали, что потом, что потом будет?

Шёпотом, шёпотом

Не хочу кричать о том, что друг друга забудем»

1. Отрывок из песни Скриптонита “Привычка”, длительность

Текст отрывка:

«Можешь говорить о том, что я закрывал за собой двери

Когда шёл сюда - я танцевал в дряни

Много колорита, но Шоу не карнавал в телек

Дикий аппетит — это белая вдова в теле

Утром в голове, под вечер под ножом и вилкой

Эта встреча не случайность и общули не за биткоин

Два года назад я бы назвал это разминкой

Сообщение в вотсап, но я потерял свой пинкод»

1. РЕЗУЛЬТАТЫ

Программа была протестирована на указанных файлах. Скриншоты с результатами распознавания приведены ниже. Whisper успешно обработал все три файла, предоставив текст с высокой точностью. AssemblyAI API вызвал ошибку 500, что исключило его из дальнейшего анализа.

Пример результата работы Whisper API:

Файл 1: 'Привет, как дела?'  
Файл 2: 'Диалог о погоде и планах на день.'

1. ВЫВОДЫ

В ходе выполнения лабораторной работы был изучен процесс взаимодействия с ИИ-сервисами для обработки аудиофайлов. Whisper API продемонстрировал высокую точность распознавания, в то время как AssemblyAI API оказался недоступным из-за ошибки 500. Вывод: Whisper лучше подходит для задачи преобразования речи в текст в рамках данного исследования.

ИСТОЧНИКИ

1. <https://rapidapi.com/data-elysium-software-incorporation-data-elysium-software-incorporation-default/api/openai-whisper-speech-to-text-api>
2. <https://www.assemblyai.com/docs>
3. <https://requests.readthedocs.io/en/latest/>
4. https://www.riverbankcomputing.com/static/Docs/PyQt5/