Проект по Интелигентни агенти с генеративен изкуствен интелект

Тема: Sound Recognizer

### Изготвен от: Богдан Миронов 0MI080002

[1](#_6bpy3s9n2oif)

[**1 Увод 2**](#_gjdgxs)

[**2 Разпознаване на Аудио/Текст 3**](#)

[**3 Проектиране 4**](#)

[**4 Реализация, тестване/експерименти 5**](#_2et92p0)

[**4.1 Използвани технологии, платформи и библиотеки 5**](#)

[**4.2 Реализация 5**](#)

[**4.3 Резултати** 5](#_qruot8i7lus5)

[**5 Заключение 6**](#)

[**6 Използвана литература 6**](#)

# **Увод**

Система, която разпознава текста от аудио записи и го класифицира като “задача”, “напомняне”, “бележка”, “събитие”.

Целта на системата е да улесни организацията на ежедневието на даден човек. Когато му се отвори нова задача, сети се за проект или го поканят на събитие, той да може бързо и лесно да опише на телефона си какво трябва да направи. След което системата разпознава и класифицира текста, което ще позволи на устройството само да запише в календар каквото е нужно.

# Разпознаване на Аудио/Текст

Автоматично разпознаване на реч (ASR) – преобразуване на говор в текст чрез невронни мрежи. Разработвани са софтуери за разпознаване на реч с различни цели, но най-често е с цел писане на записки или документи:

* **Speechnotes [1]**
* **Dictation [2]**
* **Talktyper [3]**
* **Voxsci [4]**
* **Whisper [5]**

За да се разпознава текст от алгоритми за машинно обучение той трябва да се векторизира - процес на преобразуване на текстови данни в числов формат. Съществуват различни методи за векторизация на текст:

1. **Bag of words**

Създава се речник от уникални думи в корпуса, след което всяко изречение или документ се представя като вектор, указващ присъствието или отсъствието на тези думи.

1. **TF-IDF (Term Frequency-Inverse Document Frequency):**

Отчита не само присъствието на думи, но и тяхната честота в даден документ спрямо честотата им в целия корпус. Така се намалява влиянието на общи думи и се подчертават по-значимите термини.

1. **Word Embeddings (Word2Vec, GloVe):**

Тези техники използват невронни мрежи за създаване на векторни представяния на думи, които улавят семантичните и синтактичните им свойства.

1. **Трансформър модели (BERT, GPT)**

# **Проектиране**

Проектът включва няколко основни компонента:

* **Извличане на данни**: Изтегляне и зареждане на набори от данни в различни формати (CSV, JSON, Parquet).
* **Предварителна обработка**: Преобразуване на текстовите данни в унифициран формат и създаване на етикети за класификация.
* **Обучение на модел**: Приложение на машинно обучение за създаване на модел, който може да предсказва категории текстове.
* **Анализ на резултатите**: Измерване на точността на модела и коригиране чрез балансиране на данните.
* **Интеграция на моделите в сървърна среда:** Двата модела са имплементирани, запазени и заредени в сървърна среда, която предоставя html сайт, с който потребителя да може да взаимодейства.

**Данни:**

* TOPv2 е набор от данни за семантичен анализ, предназначен за виртуални асистенти. Той обхваща 8 домейна: аларма, събитие, съобщения, музика, навигация, напомняне, таймер и време. Наборът съдържа над 180 000 анотирани примера.

Използва се за “събитие” и “напомняне”.

* Наборът от данни на Microsoft за местоположения и времена на изпълнение на задачи (MS-LaTTE) е колекция от 10 101 задачи.

Използва се за “задача”

* Наборът от данни “What’s Happening LA Calendar Dataset” е хостван от град Лос Анджелис и съдържа информация за различни събития, случващи се в града.

Използва се за “събитие”

* Корпусът “Blog Authorship” се състои от събраните публикации на 19 320 блогъри, събрани от blogger.com през август 2004 г. Корпусът включва общо 681 288 публикации и над 140 милиона думи.

Използва се за “бележка”

# **Реализация, тестване/експерименти**

## ***Използвани технологии, платформи и библиотеки***

* **Python** – основен език за обработка и анализ.
* **Pandas** – за манипулация на данни.
* **Hugging Face Datasets [6]** – за извличане на готови корпуси.
* **Whisper AI [5]** – за преобразуване на аудио в текст.
* **Flask -** за сървър.
* **Sqlite3** - за база данни.

## ***Реализация***

Проектът представлява сървърна реализация, която последователно използва два модела и връща резултата на потребителя. Така се постига система, която позволява качването на mp4 файл със запис на гласа на потребителя. След като се качи файла, той бива преработен и класифициран от моделите, след което резултата се връща на потребителя.

Преобразуването на аудио в текст се поема от готовия модел Whisper AI [5] и изхода на модела се подава на текстовия класификатор.

Основни стъпки в реализацията на текстовия класификатор:

* **Изтегляне на набори от данни**: Данните се зареждат чрез HTTP заявки или от локални файлове.
* **Обработка и обединяване на данни**: Всички текстови записи се трансформират в унифициран формат.
* **Тренировка на модел**: Създаване на тръбопровод за обработка на текстове и класификация.
* **Анализ и оценка на резултатите**: Оценка чрез метрики като точност, отчет на класификацията и матрица на объркване.

## Резултати

Класификатор:

* accuracy: 0.9764960826804467
* Confusion Matrix:

[[1453 34 3 10]

[ 3 1472 2 22]

[ 0 3 1495 2]

[ 12 47 3 1438]]

* F1-score: 0.98

# **Заключение**

Проектът предоставя работещ модел за автоматична класификация на аудио, като използва преводач от аудио към текст и класификатор на текст.

Възможни подобрения включват интеграция с външни системи за прилагане на резултатите от класификацията.

# **Използвана литература**

[1] - <https://speechnotes.co/?ref=cybrhome>

[2] - <https://dictation.io/>

[3] - <https://talktyper.com/>

[4] - <https://www.voxsci.com/>

[5] - <https://openai.com/index/whisper/>

[6] - <https://huggingface.co/>