Интелигентен детектор за кашлица



Изготвил: Александър Цветанов 4 курс софтуерно инженерство

02.07.2021

1. Функционални и нефункционални изисквания

1.1 Функционални изисквания:

Програмата трябва да може да прави записи от микрофона на устройството, да ги записва, да обучавам модел на база на събраните данни и да предвижда наличие на кашлица измежду разговора.

1.2 Нефункционални изисквания:

Програмата трябва да е имплементирана на питон и да има необходимите модули за обработка на аудио записи и работа с невронни мрежи.

2. Използвани технологии

За аудио обработка: sounddivice

За работа със csv файлове: csv

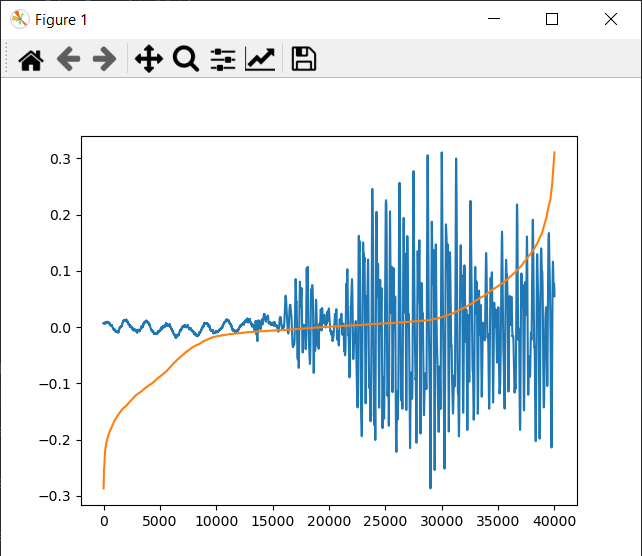
За чертаене на графики: mathplotlib

За обработка на данните: tensorflow skitlearn

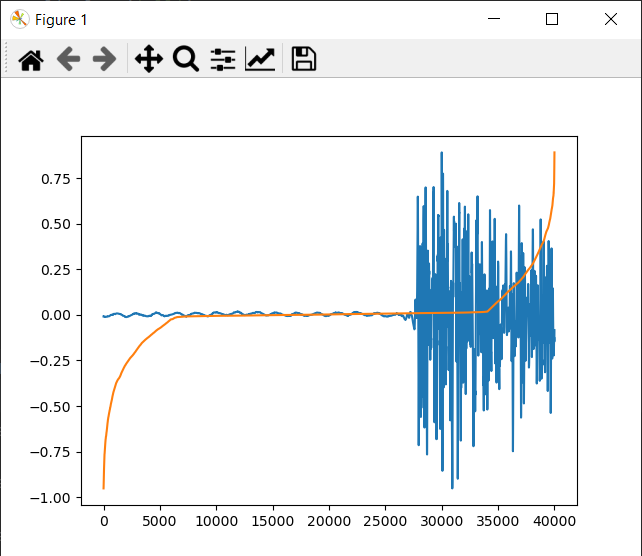
3. Нетривиални аспекти на имплементацията

Като най-нетривиален аспект от разработката на приложението бих посочил изследването на графики и подбиране то на модели и входни данни. Преди имплементирането на модела прекарах доста време във анализ на графики на звукови фрагменти за да подбера features който най-добре описват проблема. До извод стигнах, че е най-добре да се изследва интервала от 40000 записа около най-високата стойност на графиката разделени на 30000 преди и 10000 след нея. Причина за това е, че почти винаги децибелите на кашлица са по високи от тези на нормално говорене. Понеже преди кашлица се поема дъх първият ми feature беше разликата межди най-ниската и най-високата точка в този интервал. Понеже това поемане на въздух трае голяма част от секундата в този интервал също има и голяма разлика в медианата от обикновен разговор. Понеже обаче медианата на дори обикновен запис е около нулата заради отрицателните стойности на звука аз взимам медианата на 25 и 75 процент където разликата е доста по забележима.

Разговор



Кашлица



За избора на модел първо пробвах с по прости модели като линейна или логистична регресия но те нямаха голяма точност поради малкия брой данни и затова пробвах с невронна мрежа.

4. Значими елементи на архитектурата

Най-значимия елемент на архитектурата е невронна мрежа с три слоя и relu като активационна функция на първите два и sigmoid за третия.

5. Инсталация и конфигурация

Единственото нужно е да се инсталират пакетите с питонски библиотеки със pip install: sounddivice, csv, mathplotlib, tensorflow, skitlearn

6. Наръчник за потребителя

За използване е нужно само да има файла с данните в същата папка където е и кода и просто да се пусне програмата. За записване на нови данни може да се разкоментира кода в началото. След стартирана трябва източникът на звука да не по далеч от един метър. Ако се засече кашлица програмата изписва “Covid!!!”, а ако ли не периодично изписва“Everything is fine“. През определен интервал се изписва „Start“ и “Stop” което индикира, че в този интервал се записва звука.

7. Заключение

За заключени бих казал, че приложението може да се подобри с паралелна обработка, по-добри модели и повече данни. Но въпреки това смятам, че днешно време това приложения може да намери полза ако се използва в промишлен мащаб.

8. Ресурси

<https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/>

<https://python-sounddevice.readthedocs.io/en/0.4.1/>

<https://docs.python.org/3/library/csv.html>

<https://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html>

<https://towardsdatascience.com/building-a-logistic-regression-in-python-step-by->step-becd4d56c9c8