**Министерство образования и науки Российской Федерации**

**РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ НЕФТИ И ГАЗА имени И.М. ГУБКИНА**

**Кафедра информатики**

Расчетно-графическая работа по дисциплине

«Основы алгоритмизации и программирования»

тема: «Веб-приложение для распознавания окружающих звуков»

Выполнил: Луценко Н.В.

Студент группы: АЭ-20-03

Проверили:

доцент, Перепухова И.Г.

к.т.н., профессор, Сидоров В.В.

Москва 2021

# Оглавление

[Введение 3](#_gjdgxs)

[Ход работы 4](#_30j0zll)

[Заключение](#_1fob9te) 9

[Список используемых источников](#_2et92p0) 10

# Введение

Machine Learning и Deep Learning одни из главных современных трендов IT, оно находит применение во всех отраслях экономики, поэтому чтобы инженеру-программисту быть востребованным, он должен иметь опыт машинного обучения.

Цель исходной работы:

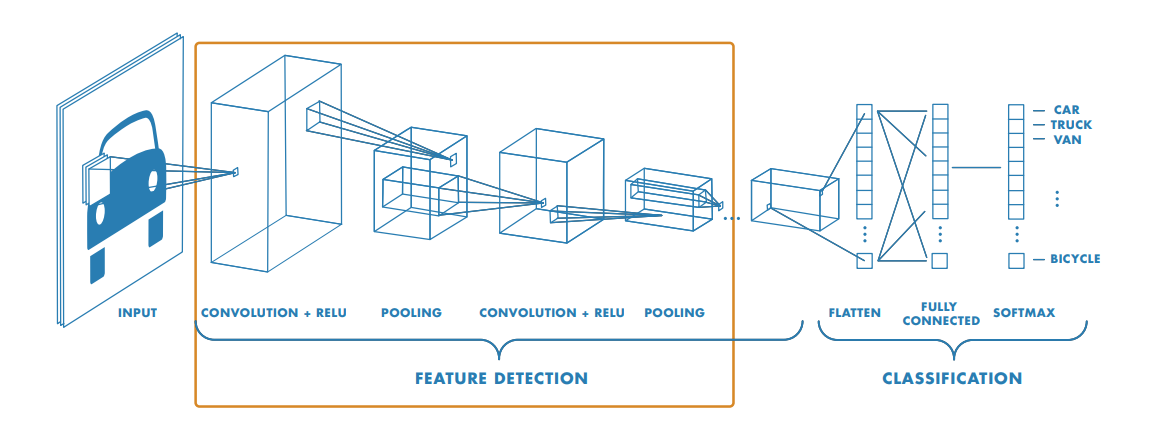
Обучить нейронную сеть и оценить ее результаты, создания веб-приложение для комфортного взаимодействия с обученной сетью.

# Ход работы

Первым делом нужно определиться, какие данные мы собираемся использовать для обучения нейронной сети и какую модель мы выбираем.

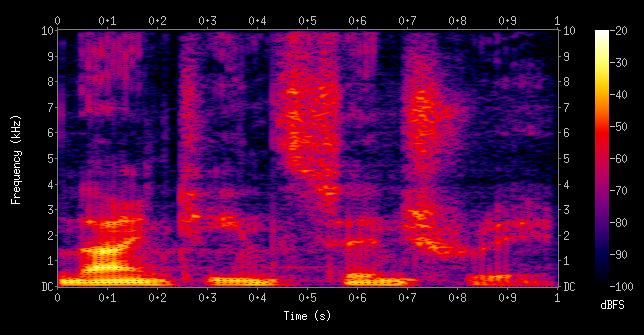
Для классификации объектов хорошо зарекомендовали себя сверточные нейронные сети. Проходя через такую сеть матрица (изображение) сжимается, то есть ее размерность, но остаются те значения, которые выявляют косвенные признаки класса, по которым в конце концов получается ответ – класс объекта.



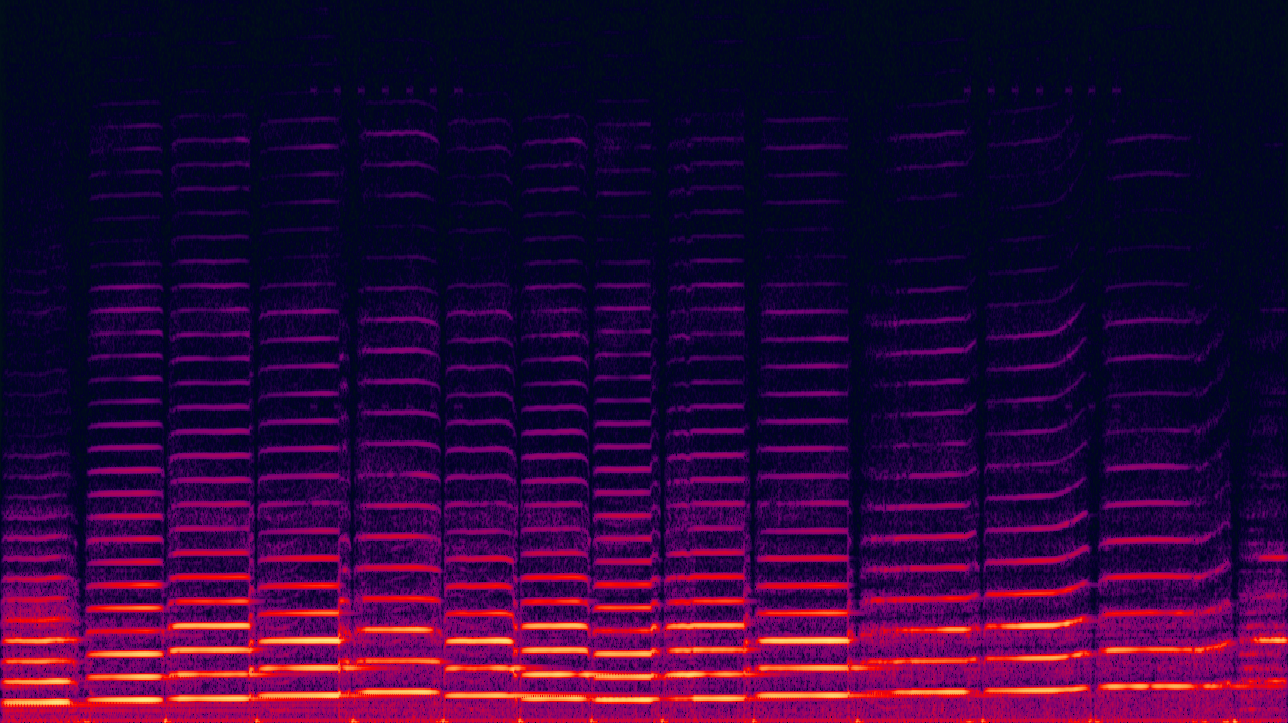


Для того, чтобы классифицировать аудиофайл с помощью сверточной нейронной сети, нужно преобразовать его в матрицу, то есть спектрограмму.

Спектрограмма — изображение, показывающее зависимость спектральной плотности мощности сигнала от времени. В каждом столбце в данный момент времени находится наблица соответствий частоты звука и его мощности



Спектрограмма мужского голоса



Спектрограмма игры на скрипки

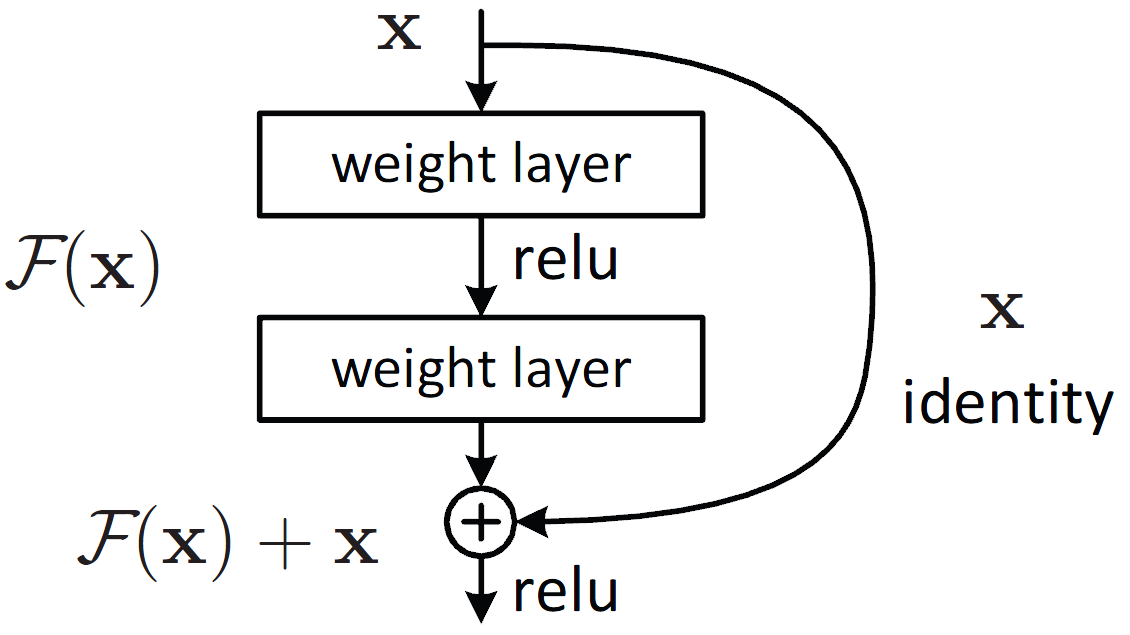
В качестве осного языка программирования выберем Python 3

Для преобрезования аудиофайлов .wav будем использовать библиотеку Librosa и matplotlib





В качестве модели нейронной сети была выбрана рекурентную сеть Resnet50

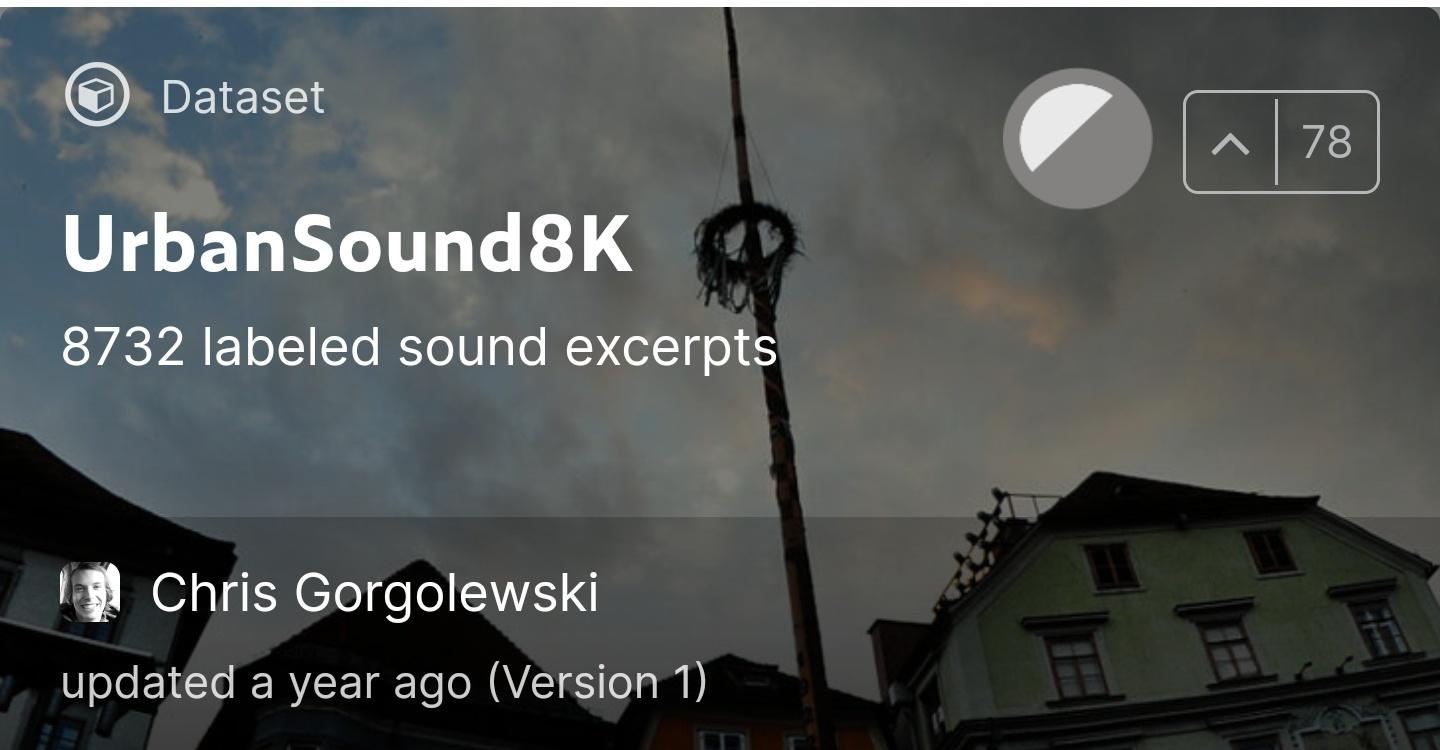


Преимуществом модели ResNet является многократная передача исходной матрицы в сеть, что позволяет "не забыть" ей первоначальную природу классифицируемого объекта, что позволяет увеличить точность распознавания

Для обучения нейронной сети была использована библиотеку Keras, являющаяся упрощённой надстройкой над фреймворком TensorFlow



Для обучения нейронной сети был использован датасетом [UrbanSound8K](https://www.kaggle.com/chrisfilo/urbansound8k), размещенным на Kaggle

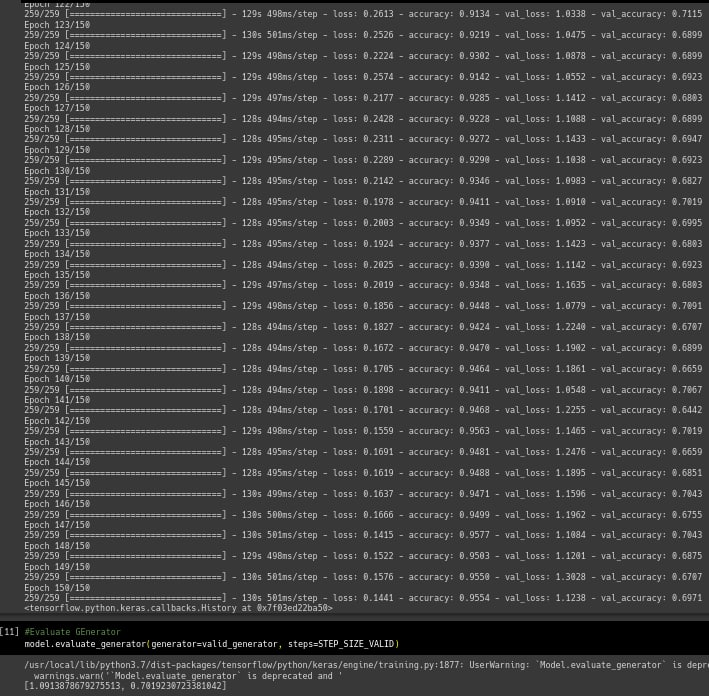


Для ускорения обучения также воспользуемся платформой Google Colab, которая безвозмездно предоставляет на ограниченное время



Для оценки точности модели будем использовать стандартную метрику accuracy

В процессе обучения было выполнено 150 эпох, которые заняли 5-6ч



Были обучены модели Resnet50, Resnet101, ResNet152, ResNetV2, но они дали одинаковую максимальную точность – 70% на тестовой выборке

Поэтому для веб-приложения была выбрана наиболее лёгкая и быстрая версия Resnet50

Для создания веб-приложения, нужен веб-сервер, который будет отвечать на запросы пользователя. В качестве фреймворка для веб-сервера был выбран Django



Для "вывода проекта в интернет" была использована программа ngrok, создающая туннель из интернета до нашего сервера.

Для контроля версий и публикации проекта использован GitHub: <https://github.com/NickElixir/UrbanSound>

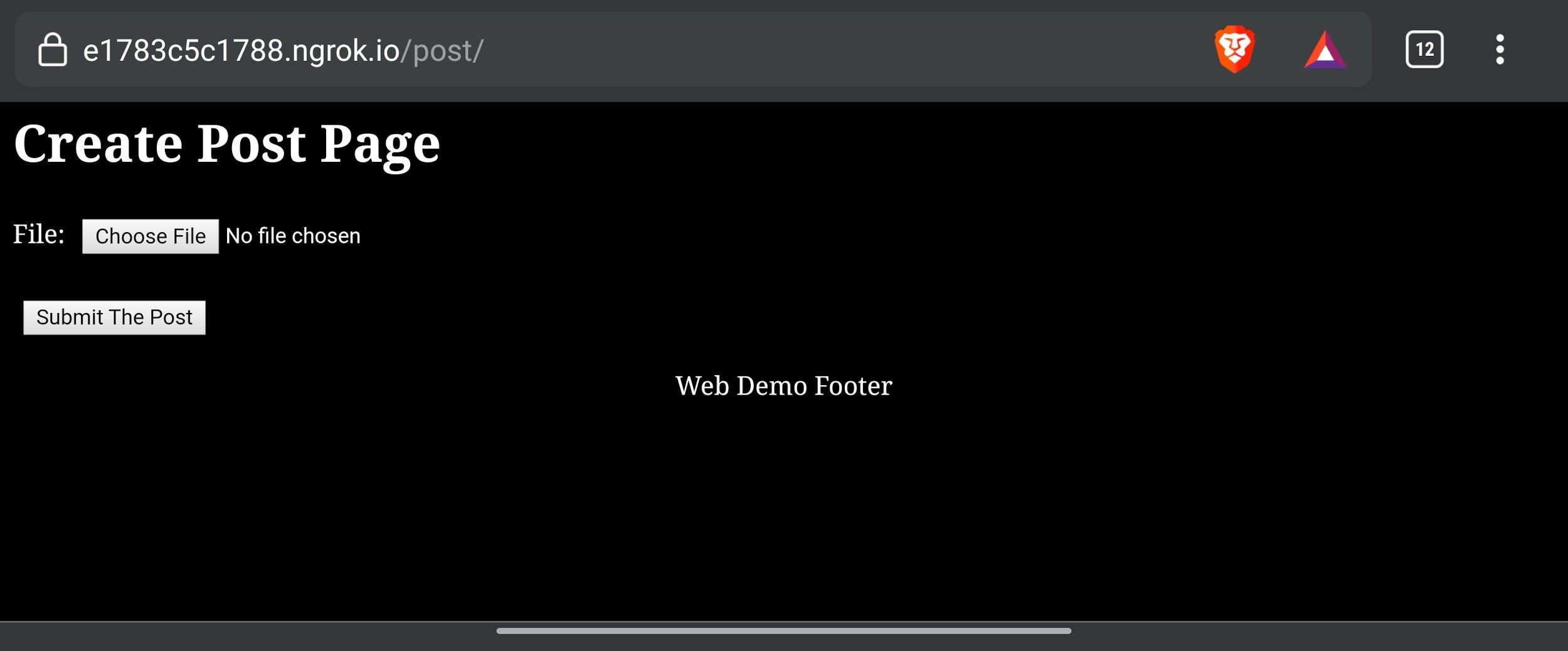


# Интерфейс программы

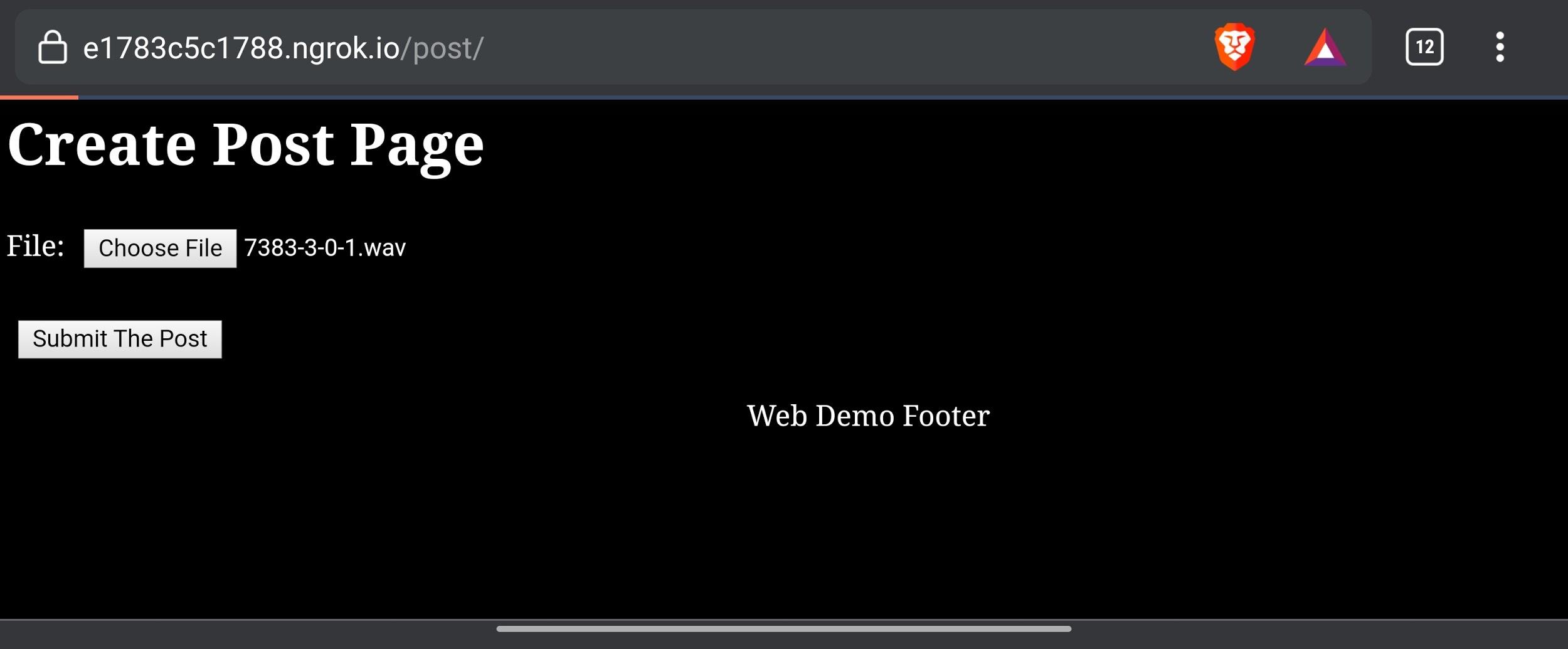
# 

На данный момент (18.05) интерфейс программы элементарный

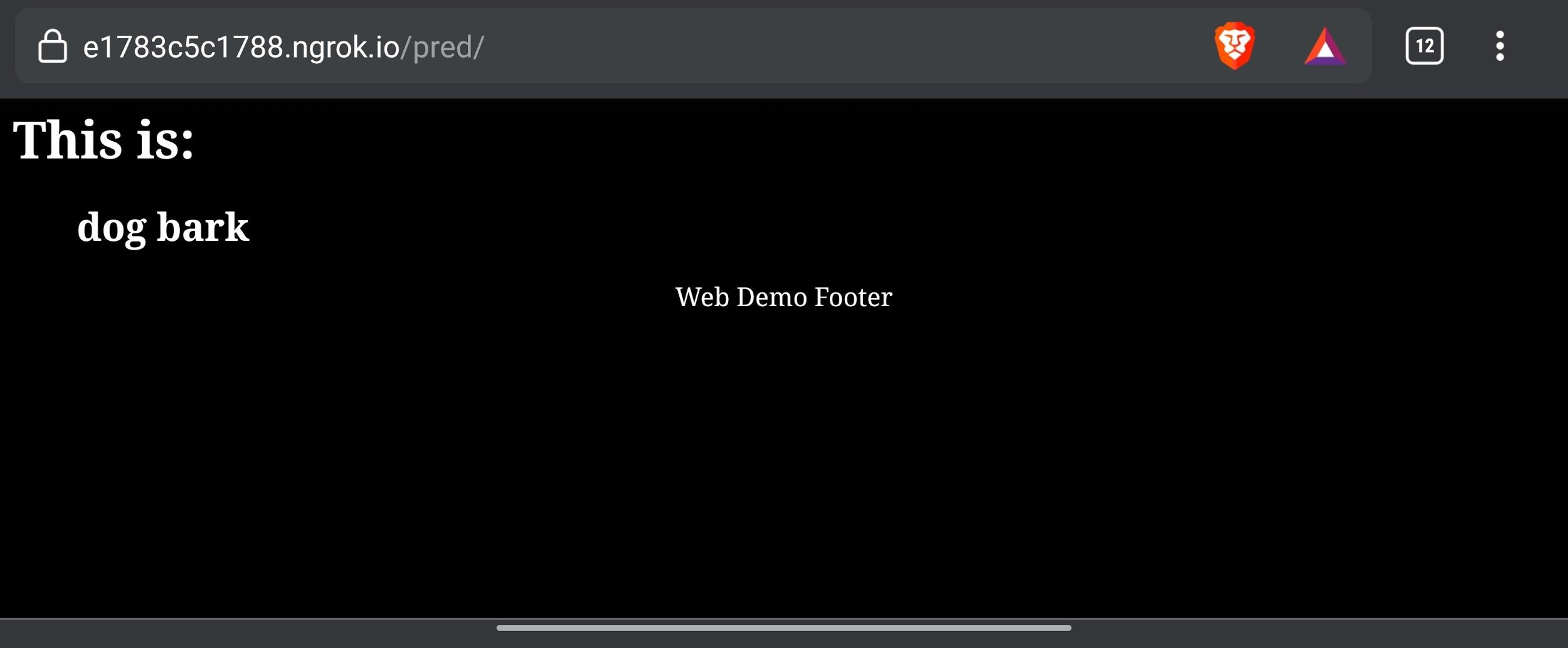
На странице /post доступна форма для загрузки аудиофайла.



Загружаем аудио с лаем собаки:



Получаем ответ на странице /pred:



# Заключение

## Что можно улучшить:

* Добавить виджет для записи аудио в браузере и отправки его на сервер
* На данный момент программа поддерживает только формат аудио wav. Необходимо добавить конвертер в wav из mp3, AAC и других на стороне сервера

## Можно ли увеличить точностью модели?

В процессе подготовки к обучению модели выбрали рассмотрены популярные методы аугментации данных – методы по увеличению различных экземпляров данных из доступных. Например, поворот изображений, обрезание, отражение, искажение, фильтры.

К сожалению, выяснилось, несмотря на то, что аудиофайл представляется в виде изображения, традиционные методы аугментации изображений не применимы для аудио, т.к. они коренным образом меняют исходные данные, теряя информацию об их изначальной принадлежности к определенным классам.

## Выводы:

Поставленная задача выполнена, опыт классификации данных с помощью Deep Learning получен. Для того, чтобы завершить проект и повысить удобство интерфейса пользователя необходимо произвести доработки.

# Список используемых источников

ResNet (34, 50, 101): «остаточные» CNN для классификации изображений URL: [<https://neurohive.io/ru/vidy-nejrosetej/resnet-34-50-101/>]

Дронов В. А. Django 3.0. Практика создания веб-сайтов на Python