



Project 4

Gebruikershandleiding

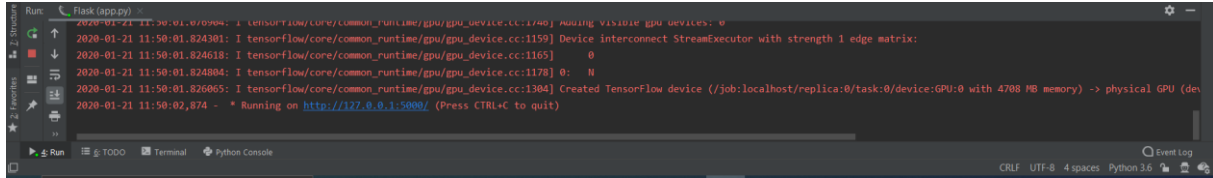
Thibault Huyghe

1. Surfen naar de interface	3
2. Gebruik maken van de interface.....	3
1. Uploaden van een geluid.....	3
2. Begrijpen van de output.....	3
3. Openen van de techinshe demo	5
4. Gebruik maken van de technische demo	5

1. Surfen naar de interface

Na het volgen van de installatiehandleiding en nadat u de code hebt gerunt met PyCharm, zou de interface actief moeten zijn. Indien de interface actief is surf dan naar:

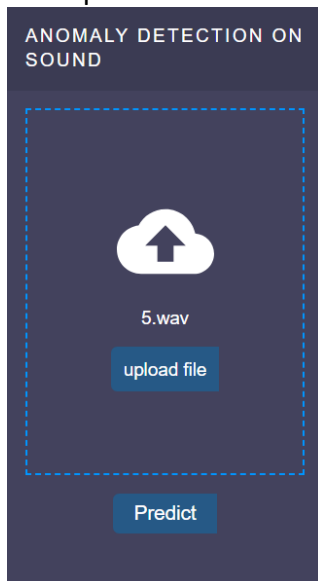
<http://127.0.0.1:5000/> of klik op de link die u in het output venster van PyCharm ziet.



2. Gebruik maken van de interface

1. Uploaden van een geluid

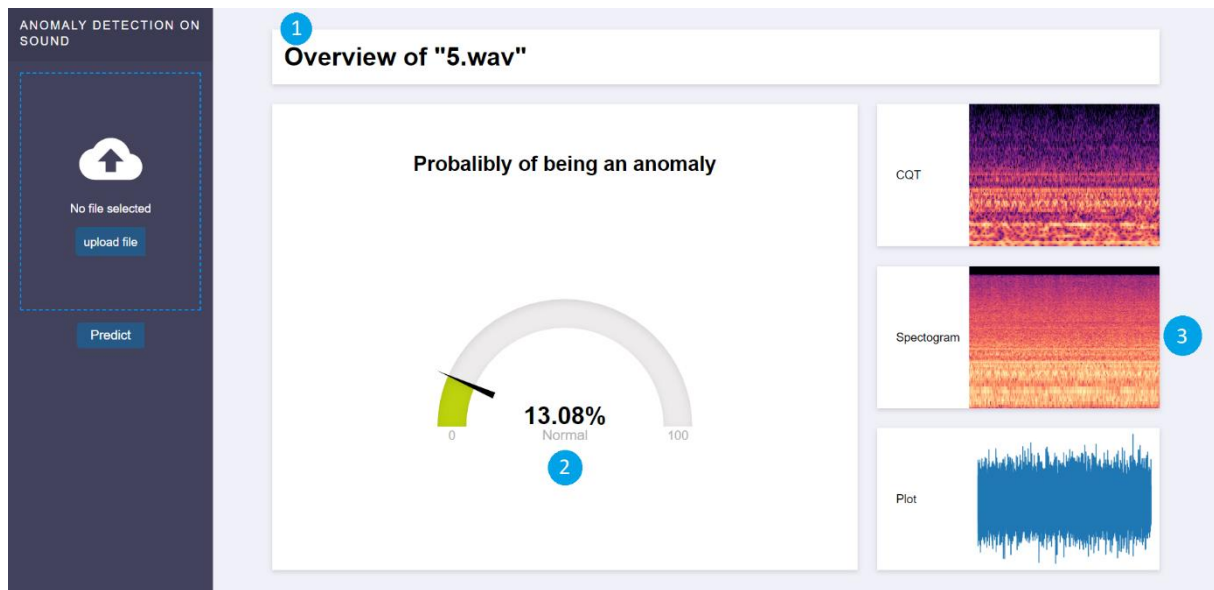
1. Eenmaal u zich op de interface bevindt, start u met het uploaden van het geluid afkomstig van de machine : "fan".
2. Upload een geluid door te klikken op "upload file" en een geluid te selecteren.
3. Voorspel de klasse van het geluid door te klikken op "Predict".



2. Begrijpen van de output

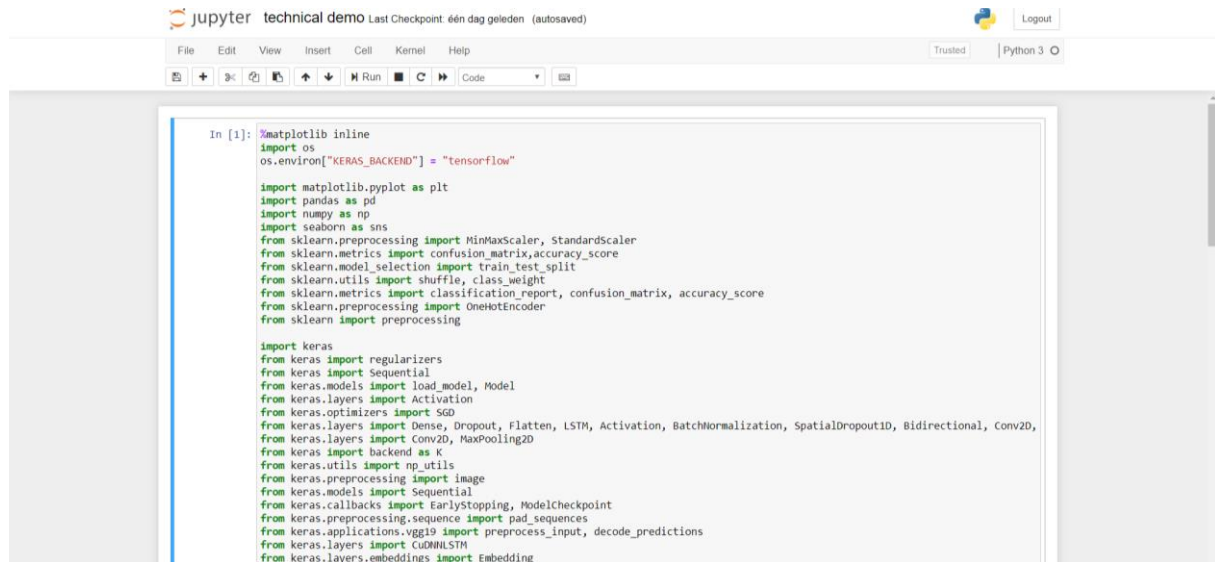
Eenmaal u een geluid hebt geüpload en geklikt hebt op "Predict" zal er verschillende informatie tevoorschijn komen:

1. Boven aan ziet u de naam van het geluid die u hebt geüpload.
2. In het midden ziet u de kans van het zijn van een anomalie en de klasse die werd voorspeld.
3. Rechts ziet u de verschillende representaties van het geluid.



3. Openen van de techinshe demo

Na het volgen van de installatiehandleiding zou u de technische demo geopend moeten hebben, indien dit niet het geval is start jupyter notebook door “jupyter notebook” in te geven als commando in cmd. Open vervolgens de demo door te navigeren naar de plaats waar u de code downloade en navigeer dan naar “model/final models/technical demo”. U zou op volgend bestand moeten terechtkomen.



```

In [1]: %matplotlib inline
import os
os.environ["KERAS_BACKEND"] = "tensorflow"

import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.preprocessing import MinMaxScaler, StandardScaler
from sklearn.metrics import confusion_matrix, accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.utils import shuffle, class_weight
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
from sklearn.preprocessing import OneHotEncoder
from sklearn import preprocessing

import keras
from keras import regularizers
from keras import Sequential
from keras.models import load_model, Model
from keras.layers import Activation
from keras.optimizers import SGD
from keras.layers import Dense, Dropout, Flatten, LSTM, Activation, BatchNormalization, SpatialDropout1D, Bidirectional, Conv2D,
from keras.layers import Conv2D, MaxPooling2D
from keras import backend as K
from keras.utils import np_utils
from keras.preprocessing import image
from keras.models import Sequential
from keras.callbacks import EarlyStopping, ModelCheckpoint
from keras.preprocessing.sequence import pad_sequences
from keras.applications.vgg19 import preprocess_input, decode_predictions
from keras.layers import CuDNNLSTM
from keras.layers.embeddings import Embedding
  
```

4. Gebruik maken van de technische demo

1. Run alle code door in de eerste cell te staan en op “run” te klikken, totdat u zich in de laatste cell bevindt.
2. De output kunt u telkens zien bij de respectievelijke cel.