**Tema 5**

**Cerinta generala a proiectului:**

Dezvoltarea unei aplicatii pentru separarea surselor unui semnal audio mono, utilizandu-se algoritmi de machine learning.

**Cerinta particulara**

Se cere separarea vocilor de zgomotul de fundal, utilizandu-se ca date de intrare in reteaua neuronala (eng. Input features) semnalul audio in domeniul timp frecventa, prin utilizarea transformatei Fourier, mai precis STFT, cu n\_fft = 1024. Functia SFTF se gaseste in cadrul bibliotecii *torch*.

Pentru procesul de antrenare se va utiliza doar magnitudinea semnalului, rezultatul trasformatei fiind un numar complex:

magnitudine= torch.sqrt(parte\_reala\*\*2 + parte\_imaginara\*\*2)

Pentru dezvoltarea solutiei se va utiliza *PyTorch*, un framework ce faciliteaza crearea si antrenarea unui algoritm de tip deep learning.

Notatii:

* speech : orice tip de voce
* noise: zgomot de fundal sau evenimente tranziente
* semnalul de intrare: mix intre speech si noise

**Seturi de date**

* **speech** : librispeech
* **noise**: TUT Acoustic scenes 2017, Development dataset, <https://zenodo.org/records/400515>
* Se cere ca frecventa de esantionare (sampling rate) a datelor sa fie de 48 KHz
* Exemplele de antrenare vor avea o durata de 10 secunde

**Componente:**

1. Procesarea datelor de intrare
   1. Generare fisiere .csv pentru noise si speech utilizandu-se *Panda DataFrames*: train, valid and test (80, 10, 10). Fiecare fisier .csv va contine o coloana index si o coloana ce contine calea catre fisier.
   2. Implementarea unui dataset loader personalizat:
      1. Implementare functie getitem() unde:
         1. Se vor selecta aleatoriu din fisierele CSV de antrenare fisiere audio pentru speech si noise
         2. Se va realiza o augmentare simpla a datelor:
            1. Normalizare speech la -30 dB
            2. Nivel SNR diferit pentru fiecare exemplu
         3. Functia getitem() va returna:
            1. Index
            2. X = noise+speech
            3. Y = speech
   3. Fisierele audio se vor deschide cu biblioteca soundfile
2. Arhitectura retelei
   1. Intrare retea: magnitudinea semnalului de intrare
   2. Iesire: estimare magnitudine speech. Pentru refacerea semnalui se va utiliza faza semnalului de intrare.
   3. Functie de cost: MSE
   4. Optimizator: Adam
   5. Structura:
      1. Unet

<https://towardsdatascience.com/unet-line-by-line-explanation-9b191c76baf5>

1. Deploy

O aplicatie simpla ce contine o interfata utilizator unde se poate incarca un fisier audio pentru a fi procesat si ascultat.

Referinte:

* <https://www.youtube.com/watch?v=TdnVE5m3o_0&list=PL-wATfeyAMNqIee7cH3q1bh4QJFAaeNv0&index=18>
* <https://pytorch.org/tutorials/index.html>