Laboratorul 4: Modele moderne de sinteză a vorbirii (TTS)

Obiective

- Diferențierea între abordările tradiționale (concatenativă, parametrică) și cele bazate pe AI pentru TTS.
- Descrierea arhitecturilor cheie în TTS modern: model acustic (ex. Tacotron) și vocoder (ex. WaveNet / HiFi-GAN).
- Înțelegerea modelelor end-to-end integrate (ex. VITS).
- Utilizarea unui model TTS pre-antrenat pentru a genera vorbire din text.
- Explorarea controlului identității vocii (multi-vorbitor / voce de referință).

Partea I: Generare de voce cu biblioteca TTS

Acest tutorial se concentrează pe utilizarea bibliotecii open-source TTS (Coqui TTS) pentru a genera vorbire din text folosind modele moderne.

Resurse necesare

- Un mediu Python cu acces la internet (recomandat: Google Colab).
- Biblioteci: torch, TTS.
- Spațiu pe disc pentru modele și fișiere .wav.

Sarcini

1. Instalarea bibliotecii:

   pip install TTS

2. Listarea modelelor disponibile (opțional, din linia de comandă):

   tts --list_models

3. Generarea vorbirii folosind un model multi-vorbitor (ex. YourTTS / VITS):

   import torch

   from TTS.api import TTS

   # Selectează dispozitivul

   device = "cuda" if torch.cuda.is_available() else "cpu"

   # Încarcă un model multi-lingvistic / multi-vorbitor

   tts = TTS("tts_models/multilingual/multi-dataset/your_tts").to(device)

   # Generează audio

   tts.tts_to_file(

text="Acesta este un test de sinteză a vorbirii în limba română.",

speaker_wav="calea/catre/voce_referinta.wav", # opțional: voce de referință (cu acord)

language="ro",

file_path="output.wav",

)

4. Experimentare (raport scurt în 5–10 rânduri):

- Schimbați propoziția (minim 3 exemple) și observați intonația și naturalețea.
- Încercați cel puțin două limbi diferite (dacă modelul le suportă).
- Dacă folosiți speaker_wav: utilizați doar vocea proprie sau o înregistrare pentru care aveți consimțământ explicit.

Partea II: Calitatea vocii

Generați mostre audio pentru aceeași propoziție folosind două abordări diferite și comparați auditiv:

- Un sistem parametric clasic (ex. eSpeak NG sau alt TTS tradițional disponibil).
- Un model neuronal modern (ex. VITS din biblioteca TTS).

În raport, discutați: naturalețea, claritatea, prozodia, artefactele și viteza de generare.

Partea III: Generarea de voci diverse (multi-speaker)

Folosind un model multi-vorbitor, generați aceeași propoziție rostită de mai mulți vorbitori diferiți.

Sarcini

5. Alegeți un model multi-speaker (ex. un model VITS antrenat pe VCTK).
6. Afișați lista de vorbitori disponibili și selectați cel puțin 5 ID-uri diferite.
7. Generați câte un fișier .wav pentru fiecare vorbitor, folosind aceeași propoziție.
8. Comparați diferențele de timbru și intonație. Includeți concluzii (minim 8–10 rânduri).

Livrabile

- Un fișier Jupyter Notebook (.ipynb) sau script Python (.py) cu pașii realizați.
- Fișierele audio generate (.wav).
- Un raport scurt (max. 1 pagină) cu observații și concluzii.

Test Formativ 4

Secțiunea I – Definiții și concepte

1) Completați tabelul de mai jos: un avantaj și un dezavantaj pentru fiecare abordare TTS.

Abordare	Avantaj	Dezavantaj
Concatenativă		
Parametrică (HMM)		
Neuronală (ex. Tacotron / VITS)		

2) Ce este un vocoder în contextul sintezei neuronale a vorbirii?

Secțiunea II – Arhitecturi și tehnologii

3) De ce modelele autoregresive precum WaveNet sunt lente la inferență, în timp ce modelele bazate pe GAN (ex. HiFi-GAN) sunt rapide? Care este compromisul?

4) Ce inovație majoră aduce un model end-to-end integrat precum VITS față de o arhitectură în două etape (model acustic + vocoder separat)?

Secțiunea III – Aplicații practice

5) Ce înseamnă sinteza zero-shot și ce este necesar pentru a o realiza cu un model precum YourTTS?