Musical Robot

Technology Review

CSE 583 Autumn 2021
Amre Abken
Olga Dorabiala
Katie Johnston
Raphael Liu

Background

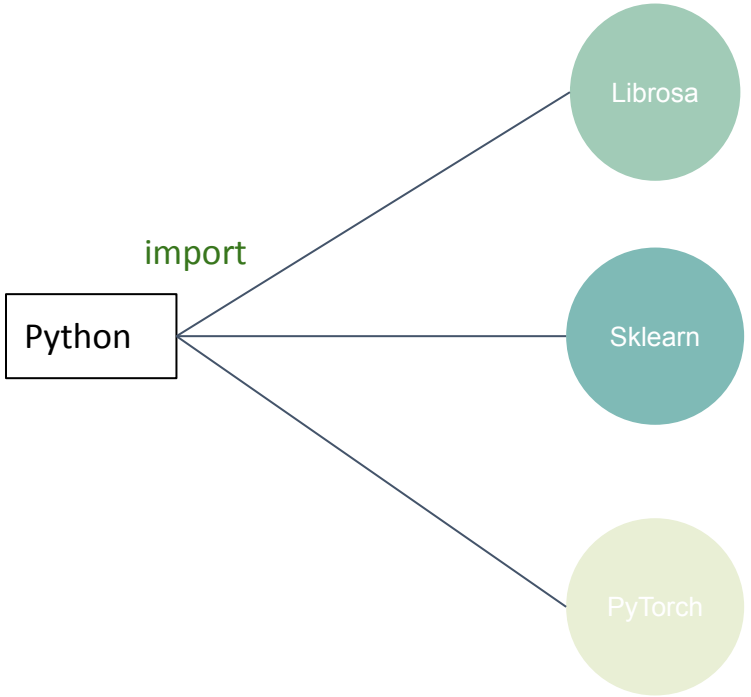
What musical robot does:

- User inputs an audio clip. The musical robot returns the genre of that song and allows the user to explore the most popular songs in the genre as well as similar genres.

What we need:

- Tools that help us process the audio clips, extract and analyze the musical features. Tools that help us build the neural network. Tools to visualize the results. Tools to query the data.

Import packages



1

Librosa

Audio and music processing tools in Python

2

Sklearn

A set of python modules for machine learning

3

PyTorch

A machine learning framework

Librosa VS. other music processing tool

How Librosa works in our project:

- Perform melody (Mel) spectrogram decomposition and waveform analysis
- Visualize spectrograms and waveforms
- Compute musical features for dataset training

Other music processing tools include but not limited to:

Libfmp — a more educational package, targeting to researches with higher demand of functionality

Aubio — detect musical events from audio signals, written in C and has a Python interface.

`

Why Librosa?

— Well established and widely used since 2015 , and constantly updated
— Compatible to most audio files like MP3
— Convenient to use with expected functionality

McFee, Brian, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. “librosa: Audio and music signal analysis in python.” In Proceedings of the 14th python in science conference, pp. 18-25. 2015.

Scikit - Learn

Need:

- Some preprocessing of data before training
  - train and test data split
  - OneHotEncoding because we have numerically labeled data
- Clustering (to group similar songs together)
  - K nearest neighbors, more complex clusterings

Advantages/ Reasons to pick:

- Very well known library
  - Many examples and tutorials on the internet
  - We have used it before
- Has a very diverse array of tools
  - Can help limit the number of packages we need
  - Tools will work together better

Alternatives:

- Preprocessing with pytorch
  - Still a possibility
  - Sklearn is a little more user, and has better tutorials
  - Group past experience
- Code ourselves,
  - Clustering, preprocessing
    - Don’t reinvent the wheel
    - Higher possibility of bugs
- When googling clustering in python, only found different types of sklearn clustering.
- Using knn, and then probably more complex clustering algorithm dependent on outcomes

- A machine learning framework produced by Facebook in October 2016
- Based on the popular Torch library

The reasons why we use PyTorch:

- Easy to learn
- Strong community
- Easy debugging
- Can accelerate computing using GPU
- Very flexible

Alternatives:

- **Sklearn**
  - Easy to use, but slower and not designed for deep neural networks
- **TensorFlow**
  - Harder to use than PyTorch
  - We have less experience with TF
- **Keras**
  - Slower than PyTorch
  - Works best with smaller datasets
  - Now a part of TensorFlow

Thank You!