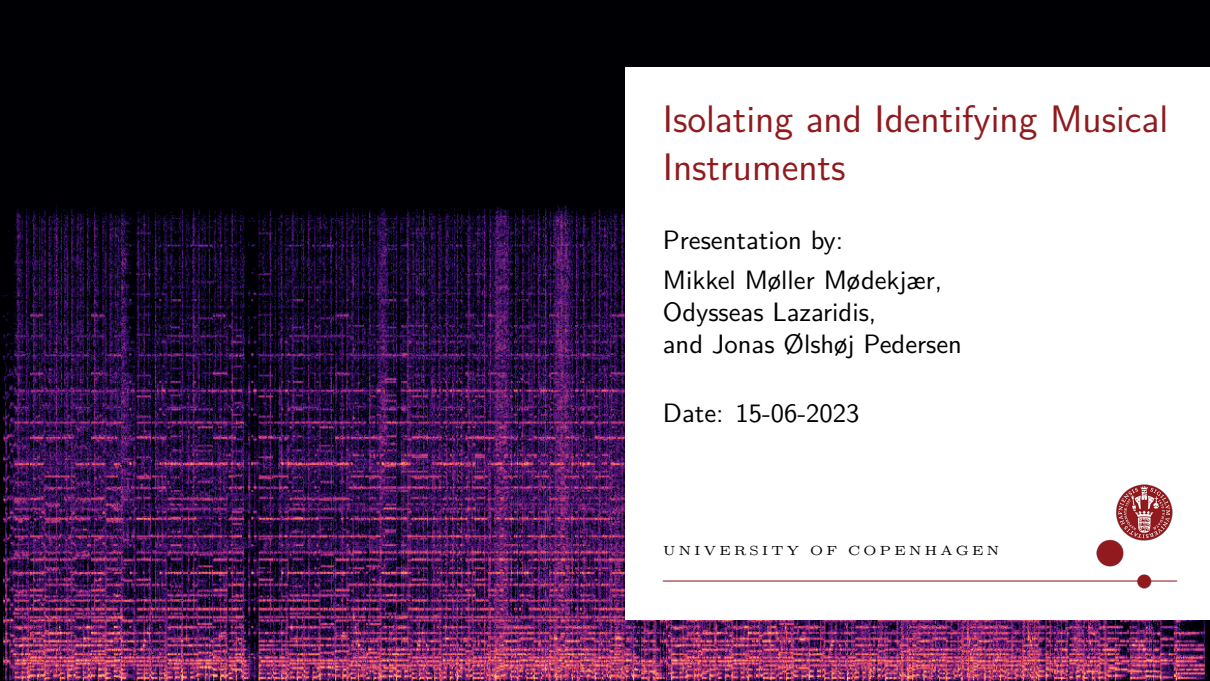


Isolating and Identifying Musical Instruments

Presentation by:

Mikkel Møller Mødekjær,
Odysseas Lazaridis,
and Jonas Ølshøj Pedersen

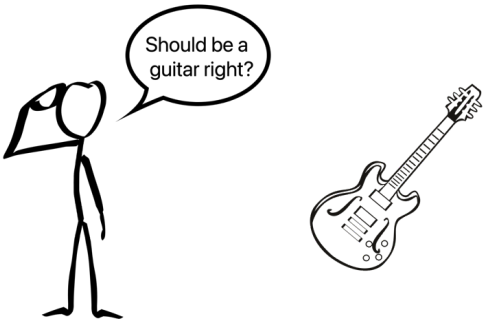
Date: 15-06-2023

UNIVERSITY OF COPENHAGEN

Tasks

Isolating Individual Instruments

Classifying Instruments

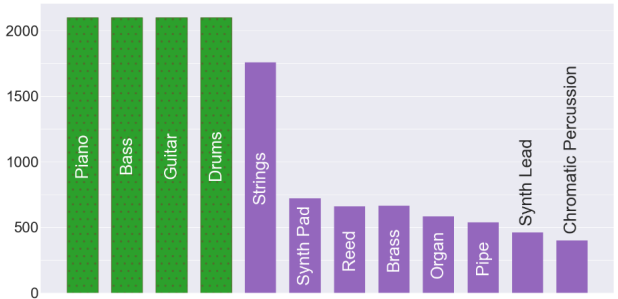


Musical Analysis

Slakh2100

The Slakh dataset consists of 2100 songs (≈100 GB), already with instruments isolated.

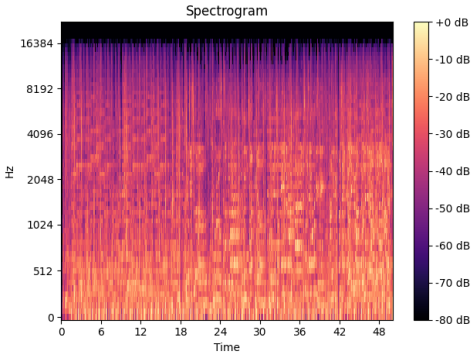
- All songs given in synthetic MIDI tracks.



http://www.slakh.com/

Short-Time Fourier Transform (STFT)

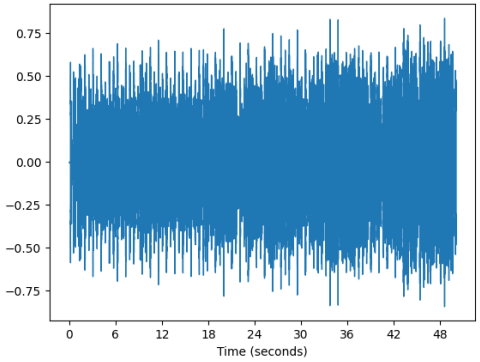
- A way to analyze the frequency content of a signal over time.
- Window Size: Determines the length of the window in which we apply each individual Fourier Transformation
- Time and frequency resolution trade-off.
- Note, STFT gives complex values



Isolating Instruments

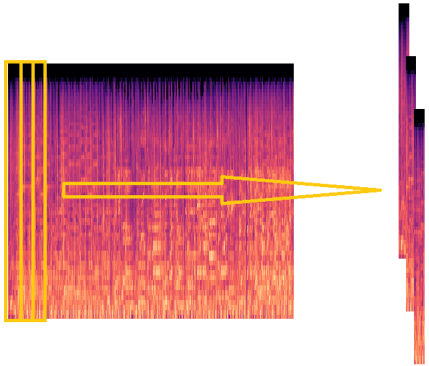
The Problem we try to solve

- We try to isolate individual instruments from what you see on the picture on the right.
- How can we get more information about the song?



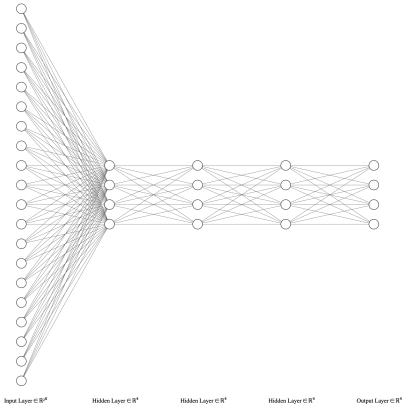
Data Processing

- We use an STFT to transform the mixture signal into the frequency domain
- We built the input vector $x \in \mathbb{C}^{(2C+1)L}$ by stacking the magnitude values of the C surrounding frames ($C = 3$).
- For the training we used 100 random snippets from each song



DNN architecture

- The Input Layer is made out of 7 music frames (of length 129) stacked. This makes an input of 903 neurons.
- All layers are linear.
- The output should be one STFT-frame of the isolated instrument
- The network is scaled down by 32 times in order to fit in the picture

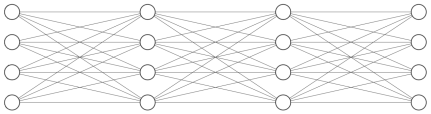


Weight Initialization

The structure of our network allows us to train each layer individually

- All weights can be initialized by a least squares solution
- This allows for faster convergence
- Relatively small computation cost

...or we could just initialize random weights (we did both)



$$\mathbf{W}_k^{\text{init}} = \mathbf{C}_{sx}\mathbf{C}_{xx}^{-1}, \qquad \mathbf{b}_k^{\text{init}} = \bar{\mathbf{s}} - \mathbf{W}_k^{\text{init}}\bar{\mathbf{x}}_k,$$

with

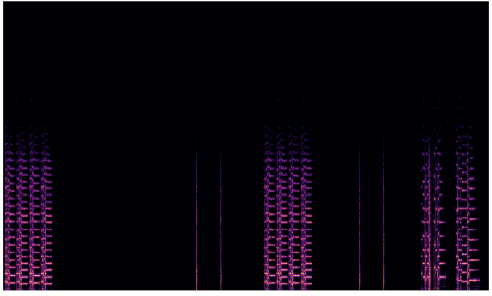
$$\mathbf{C}_{sx} = \sum_{p=1}^{P} \left(\mathbf{s}^{(p)} - \bar{\mathbf{s}}\right)\left(\mathbf{x}_k^{(p)} - \bar{\mathbf{x}}_k\right)^T,$$

$$\mathbf{C}_{xx} = \sum_{p=1}^{P} \left(\mathbf{x}_k^{(p)} - \bar{\mathbf{x}}_k\right)\left(\mathbf{x}_k^{(p)} - \bar{\mathbf{x}}_k\right)^T,$$

Live Demonstration

Soooooo Issues...

- We used SSE, which did not affect the majority of our errors, as these were below 1 (using SAE improved results occasionally)
- There was far more background than signal
- Lack of library support for complex values in neural networks

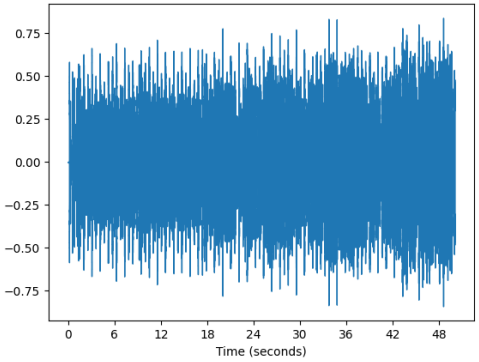


Classifying Instruments

(The good part)

The Problem we try to solve

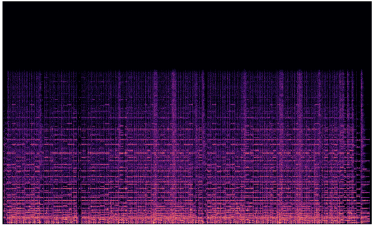
- We try to classify individual instruments from what you see on the picture on the right.

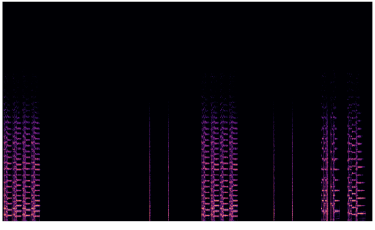


Data Processing

Starting from the STFT of each instrument from each song:

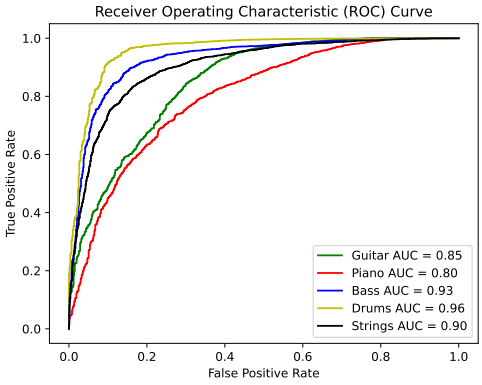
- Splitting the spectrogram into 1 second snippets.
- Label the snippets according to whether the instrument is playing or not.
- Fed the mix snippets and the labels to LightGBM.





Results

- Trained on 100 tracks (≈20000 samples), validated on 20
- Low frequencies are most recognizable
- Results are in agreement with previous work (https://doi.org/10.3390/s22083033)



Conclusion

Music analysis is hard ¯_(ツ)_/¯

Ongoing Ideas - AI Generated Tracks

Using the same Neural Network structure as previously, filling in instruments should be possible.

- The input would be the song without the instrument in question, while the output would be the full song
- Could start from a single instrument and then build up an AI generated song

Ongoing Ideas - Clustering (genres?)

From the STFT spectogram, a series of features can be extracted, such as

- The Band energy ratio which measures how dominant low frequencies are
- Spectral Centroid which show the frequency band in which we have most of the energy concentrated

With these features it can be possible to predict the music genre of a piece of music

Thank you for your time :3