Outline

- Preliminaries
- A brief history and overview of MIDI
- What kinds of information do MIDI files contain?
- Using `pretty_midi` to extract information from MIDI files
- Aligning MIDI files to audio recordings
- Matching MIDI files to audio corpora
- Exploring the Lakh MIDI Dataset
- How reliable is the LMD?
- Calls to action
If you want to follow along...

- Download “LMD mini”:
  http://colinraffel.com/projects/lmd/lmd_mini.tar.gz

- Get a Python environment set up, with (all `pip` installable)
  - pretty_midi
  - librosa
  - mir_eval
  - matplotlib
  - numpy
  - jupyter
  - tables
  - pyfluidsynth (optional, requires fluidsynth)
Why am I doing this?
And what are you going to get out of it?
MIDI FILES

MIDI FILES EVERYWHERE
What is MIDI?

Photo by user Pretzelpaws, Wikipedia, CC-BY-SA
What is MIDI?
MIDI Files

- Change to program number 3
- Play note 60 (C4) at velocity 64
MIDI Files

MIDI Data

Instrument 1: Piano

Instrument 2: Cello
General MIDI

Photo by user Darashinaikuma, Wikipedia, CC-BY-SA
## MIDI BBS

<table>
<thead>
<tr>
<th>Company</th>
<th>Location</th>
<th>Name</th>
<th>Phone Number</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>MystMagical MIDI</td>
<td>Omaha, NB</td>
<td>Pete Olsen</td>
<td>402-293-0451</td>
<td></td>
</tr>
<tr>
<td>PGH-MIDI Music</td>
<td>Pittsburg, PA</td>
<td>Art Doud</td>
<td>412-882-3703</td>
<td></td>
</tr>
<tr>
<td>MIDI Thru</td>
<td>Aveiro, PORT</td>
<td>Fausto Carvalho</td>
<td>35-1-34915452</td>
<td></td>
</tr>
<tr>
<td>1st Dutch MIDI</td>
<td>Delft, HOLLAND</td>
<td>Johan Corstjens</td>
<td>31-1-5138754</td>
<td></td>
</tr>
<tr>
<td>TBUS-BBS</td>
<td>Munich, GERMANY</td>
<td>Rudolf Stricker</td>
<td>49-8-9293881</td>
<td></td>
</tr>
<tr>
<td>1st Austria MIDI</td>
<td>Vienna, AUSTRIA</td>
<td>Erich Varga</td>
<td>43-1-7693132</td>
<td></td>
</tr>
<tr>
<td>Music Studio UK</td>
<td>UNITED KINGDOM</td>
<td>Paul Urmston</td>
<td>44-0+926403904</td>
<td></td>
</tr>
<tr>
<td>Rock &amp; Jazz</td>
<td>Paris, FRANCE</td>
<td>Sam Przyswa</td>
<td>33+1-40548604</td>
<td></td>
</tr>
<tr>
<td>Slatch</td>
<td>Paris, FRANCE</td>
<td>Frank Gardes</td>
<td>33+1-48020814</td>
<td></td>
</tr>
<tr>
<td>Twilight Zone</td>
<td>Barrie, ONTARIO</td>
<td>Robin Wells</td>
<td>705-722-8184</td>
<td></td>
</tr>
<tr>
<td>Action Link</td>
<td>Bradenton, FL</td>
<td>Jim Davie</td>
<td>813-747-9295</td>
<td></td>
</tr>
</tbody>
</table>

"0" after the telephone number means that BBS has a door for off-line message handling. Saves time on line, and $ long distance!
“Bulletin Board Systems”

Photo by user Liftarn, Wikipedia, CC-BY-SA
“You log on to your computer service (Compuserve, GEnie, Prodigy, America On Line, etc.), and you see that someone has made a MIDI Song File of your song, and has uploaded it so any Tom, Dick, or Harry can download it. You even download a copy out of curiosity. As they don't charge you any extra to download it, you assume that you'll get paid royalties out of the online flat-rate fee. Wrong! The MIDI File of your song is being given away (yes - published electronically for free), and no royalties are being collected or distributed.”
Why are there so many of them?

“Prediction: The cost for 128 kilobytes of memory will fall below U$100 in the near future.”

Creative Computing magazine
December 1981, page 6
Why are there so many of them?

Photo by user Pokman817, Wikipedia, CC-BY-SA
Why are there so many of them?
What information is typically in MIDI files?
Lengths

![Histogram of MIDI file lengths](image-url)
Transcription
Instrumentation
Tempo

![Histogram of MIDI files by number of tempo changes](image.png)
Tempo

![Histogram of tempo occurrences](image)

- X-axis: Tempo (in beats per minute)
- Y-axis: Thousands of occurrences
Time Signature
Time Signature
Key Signature
Key Signature
Lyrics
Lyrics
Things you can’t (reliably) get from MIDI

- Vocal track
- Melody track
- Instrument name
- Chord annotations
- Structural annotations
- Metadata
pretty_midi tutorial

https://goo.gl/YL687S
Matching

artist: 'Tori Amos'
release: 'LIVE AT MONTREUX'
title: 'Smells Like Teen Spirit'
id: 'TRKUYPW128F92E1FC0'
duration: 216.4502
sample_rate: 22050
audio_md5: '8'
7digitalid: 5764727
year: 1992
Matching by metadata won’t work

- J/Jerseygi.mid
- V/VARIA180.MID
- Carpenters/WeveOnly.mid
- 2009 MIDI/handy_man1-D105.mid
- G/Garotos Modernos - Bailanta De Fronteira.mid
- Various Artists/REWINDNAS.MID
- GoldenEarring/Twilight_Zone.mid
- Sure.Polyphone.Midi/Poly 2268.mid
- d/danza3.mid
- 100%sure.polyphone.midi/Fresh.mid
- rogers_kenny/medley.mid
- 2009 MIDI/looking_out_my_backdoor3-Bb192.mid
Sequence Retrieval
Dynamic Time Warping
Dynamic Time Warping
Representation?

Chromagram?

Constant-Q Spectrogram?

Log Magnitude?

Z-scored?

L2-normalized?

Beat-synchronous?
Path constraints?
Score normalization?

\[
\text{score} = \frac{\left| \mathbf{p}_m \right|}{\sum_{i=1}^{\left| \mathbf{p}_m \right|} D[\mathbf{p}_m[i], \mathbf{p}_a[i]] + \Phi(i)} \frac{\max(p_m) \max(p_a)}{\sum_{i=\min(p_m)}^{\max(p_m)} \sum_{j=\min(p_a)}^{\max(p_a)} D[i,j]} \frac{\left| \mathbf{p}_m \right|}{\max(p_m) - \min(p_m)} \frac{\max(p_a) - \min(p_a)}{\max(p_a) - \min(p_a)}
\]
Bayesian Optimization

Expected Improvement
Idea: Synthetic Alignment Data

Original MIDI CQT

After corruption
Artificial time warping

![Graph showing time warping applied](image)
Correcting time warping

Timing correction

Offset from original time vs. Original time

- Ground-truth offset
- Fixed corrupted offset
Measuring error
Score normalization search
Best system

- Use log-magnitude constant-Q spectrograms
- Don’t beat synchronize
- L2 normalize spectra (cosine distance)
- Don’t z-score spectrograms
- Use median distance as non-diagonal penalty
- Force sequences to match up to 96% of shorter
- Don’t use a band path constraint
- Include penalties in confidence score
- Normalize by path length and submatrix mean
- Example implementation in pretty_midi examples
Real-world test
DTW is too slow

(.247 seconds)(178,561 MIDI files)(994,960 audio files)

(60 seconds)(60 minutes)(24 hours)(365 days)
DTW is too slow

\[
\frac{.247 \text{ seconds}}{60 \text{ seconds}} \cdot \frac{178,561 \text{ MIDI files}}{60 \text{ minutes}} \cdot \frac{994,960 \text{ audio files}}{24 \text{ hours}} \cdot \frac{365 \text{ days}}{365 \text{ days}} \approx 1,391 \text{ years}
\]
Downsampled Hash Sequences

\[
\text{POPCNT}(\begin{array}{c}
\begin{array}{cccccccc}
1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
\end{array}
\end{array}\end{array}) = 9
\]
Similarity-preserving hashing
Collecting data

- MIDI: 140,910
- Hand: 24,850
- Clock: 17,243
- Rocket: 26,311
- Grid: 10,035

Pie chart:
- Development: 25%
- Validate: 10%
- Test: 25%
- Train: 40%
Network structure
Loss function

\[ \mathcal{L} = \frac{1}{|\mathcal{P}|} \sum_{(x,y) \in \mathcal{P}} \| f(x) - g(y) \|_2^2 + \frac{\alpha}{|\mathcal{N}|} \sum_{(x,y) \in \mathcal{N}} \max(0, m - \| f(x) - g(y) \|_2)^2 \]
Example output
Raw distance distributions
Network output distance distributions
Hash distance distributions
Match ranks
Sequence embedding
Pairwise sequence embedding
Pairwise sequence embedding
Feedforward attention
Embedding network
Loss function

$$\mathcal{L} = \frac{1}{|\mathcal{P}|} \sum_{(x, y) \in \mathcal{P}} \|f(x) - g(y)\|_2^2 + \frac{\alpha}{|\mathcal{N}|} \sum_{(x, y) \in \mathcal{N}} \max(0, m - \|f(x) - g(y)\|_2)^2$$
Example embeddings
Example attention
Attention dreaming
Embedding distances
Match ranks
Combined ranks

![Graph showing combined ranks for different categories.](image-url)
Number of matches
Lakh MIDI Dataset (mini) tutorial

https://goo.gl/hU4GAK
How well did we do?

- Incorrect match
- Correct match, sloppy alignment
- Good to go
Decoys?

- Kaskade - In This Life (Mario Fabriani Remix)
- Marcos Hernandez - If You Were Mine
- D-Unity - Afrika
- Anane - Let’s Get High (Yves C Mix)
- DJ Silver - Wardance
- Ann Nesby - So Much Joy (Praise Party Beats)
- Kim English - My Destiny (Kobbe & Austin Leeds Club Mix)
- Gianluca Motta / Snap! / NG3 - Ooops Up
- Dave Clarke - Protective Custody
- Karizma - Ride (Original Mix)
- Logic - The Warning (Claude Monnet & Torre Bros Main Mix)
- Ultra Nat - Automatic (Shawn Q’s Soltribe Vocal Mix)
More examples
Decoys
Testing key reliability

→ C major

→ C major

→ G major
### Key Results

<table>
<thead>
<tr>
<th>Source</th>
<th>Score</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIDI, all keys</td>
<td>0.400</td>
<td>223</td>
</tr>
<tr>
<td>MIDI, C major only</td>
<td>0.167</td>
<td>146</td>
</tr>
<tr>
<td>MIDI, non-C major</td>
<td>0.842</td>
<td>77</td>
</tr>
<tr>
<td>Audio content-based</td>
<td>0.687</td>
<td>151</td>
</tr>
<tr>
<td>Human Agreement</td>
<td>0.857</td>
<td>145</td>
</tr>
</tbody>
</table>
Testing beat reliability

Confidence: 0.87
Testing key reliability
Beat results
Idea: Improve DTW confidence score reporting

- Confidence score seems somewhat invariant to sloppiness
- Also is not reliable for covers, silence/garbage
- This is the final step for matching, so it better be good!
- Can we take a learning-based approach?
- Annotate (more) bad match/bad alignment/good to go
- Train a classifier on (features of) the alignment
- Preliminary results are promising
Idea: Improve/replace DTW

- DTW itself also has some failure modes...
- ...but we did a pretty exhaustive search.
- Can we improve its fine-grained temporal alignment?
- What if the features are better? We saw that network output distance distributions were much nicer.
- Can we combine many existing alignment approaches?
- Can we learn the alignment algorithm?
Idea: Better pruning

- The pairwise sequence embedding approach is simplistic
- Feedforward attention is explicitly order invariant
- Recent results show using a position encoding is helpful
- The network architecture search preferred a small network
- Some recent advances for training recurrent models on long sequences without running out of memory
- Better loss functions for encouraging good embeddings
Idea: Getting chord annotations from MIDI

- MIDI files don’t have chord annotations
- Chords are easier to estimate from MIDI chromagrams
- Can even separate out the guitar instruments!
- Can we transfer chord labels from aligned MP3s to the MIDI, and train as usual?
- What other things about MIDI can we exploit?
- Will the resulting estimated chords be ~human level?
Idea: Detecting whether C major key is true

- We saw that many MIDI files are erroneously given C key.
- Simple idea: Train a C major vs. not C major model.
- Also, MIDI is way easier to estimate key from!
- Can easily get pitch class histogram and transition matrix.
- Do these benefits make the resulting key annotations as reliable as human labelling?
Idea: Does this MIDI file have vocals transcribed?

- Many MIDI files are karaoke files
- Even if we get a good match and alignment, it’s not a perfect transcription if an instrument is not transcribed
- Can we create a “vocal instrument” classifier?
- Simple heuristics will probably work (monophonic, in the vocal note range)
- Can also look at program number, instrument name
- Similar ideas for finding “melody” tracks
Idea: Vocal embellishment correction

- For MIDIs with vocals, there are often embellishments
- This also makes the transcription imperfect (but maybe who cares, it’s still gives us what we want)
- Can we correct this? E.g. given vocal instrument from MIDI, do fine-grained alignment
- May help to have rudimentary vocal extraction
- Can be viewed as an informed transcription problem
Idea: Do things with the dataset

- Some things don’t require matching or aligning (training generative models, corpus studies...)
- Matching can gives more metadata in both directions
- With good alignments: Transcription, instrument activation, key changes, (down)beat tracking, lyrics transcription, tempo estimation, ...
Thanks!