

Introduction

MusPy is an open source Python library for symbolic music generation. It provides essential tools for developing a music generation system, including dataset management, data I/O, data preprocessing and model evaluation.

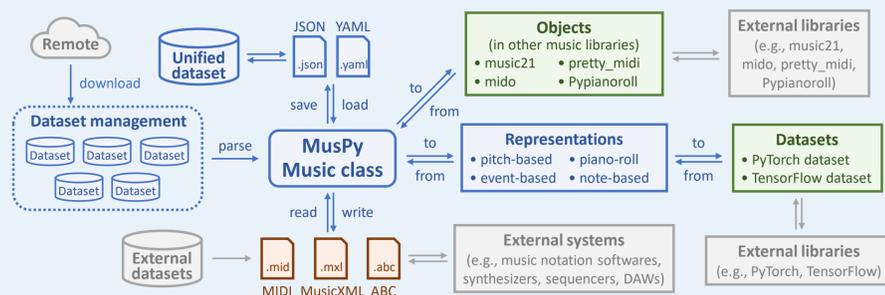


Figure 1: System diagram of MusPy.

MusPy Music Class

- Core class of MusPy
- Universal container of symbolic music
- Serializable to JSON/YAML format

I/O Interfaces

- Native JSON and YAML formats
- Common symbolic music formats
- Other symbolic music libraries
- Common representations

Datasets

- Simple commands for downloading and building a dataset
- Interfaces to PyTorch and TensorFlow

Dataset	Format	Hours	Songs	Genre	Melody	Chords	Multitrack
Lakh MIDI Dataset (LMD) [26]	MIDI	>9000	174,533	misc	△	△	△
MAESTRO Dataset [27]	MIDI	201.21	1,282	classical			
Wikifonia Lead Sheet Dataset [28]	MusicXML	198.40	6,405	misc	✓	✓	
Essen Folk Song Database [29]	ABC	56.62	9,034	folk	✓	✓	
NES Music Database [30]	MIDI	46.11	5,278	game	✓		✓
Hymnal Tune Dataset [31]	MIDI	18.74	1,756	hymn	✓		
Hymnal Dataset [31]	MIDI	17.50	1,723	hymn	△		△
music21 Corpus [24]	misc	16.86	613	misc	△		△
Nottigham Database (NMD) [32]	ABC	10.54	1,036	folk	✓	✓	
music21 JSBach Corpus [24]	MusicXML	3.46	410	classical			✓
JSBach Chorale Dataset [11]	MIDI	3.21	382	classical			✓

Table 1: Comparisons of datasets currently supported by MusPy.

Analysis Tools

- Evaluation metrics
- Score and piano-roll visualizations
- Audio rendering

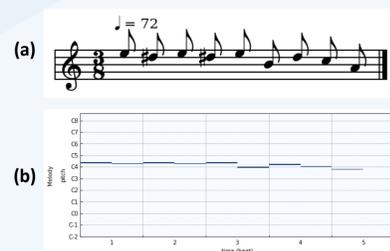
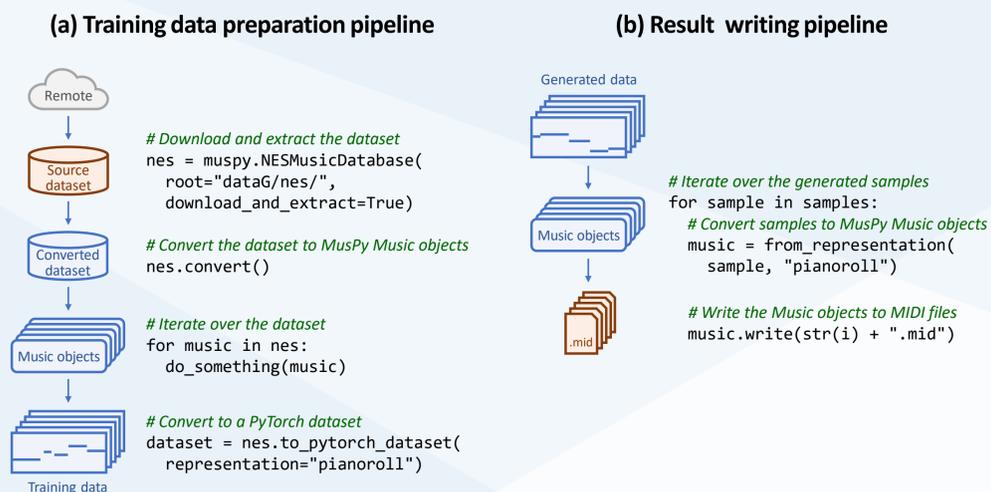


Figure 2: Examples of (a) score and (b) piano-roll visualizations.

Examples



Want to give MusPy a try?
pip install muspy

Dataset Analysis

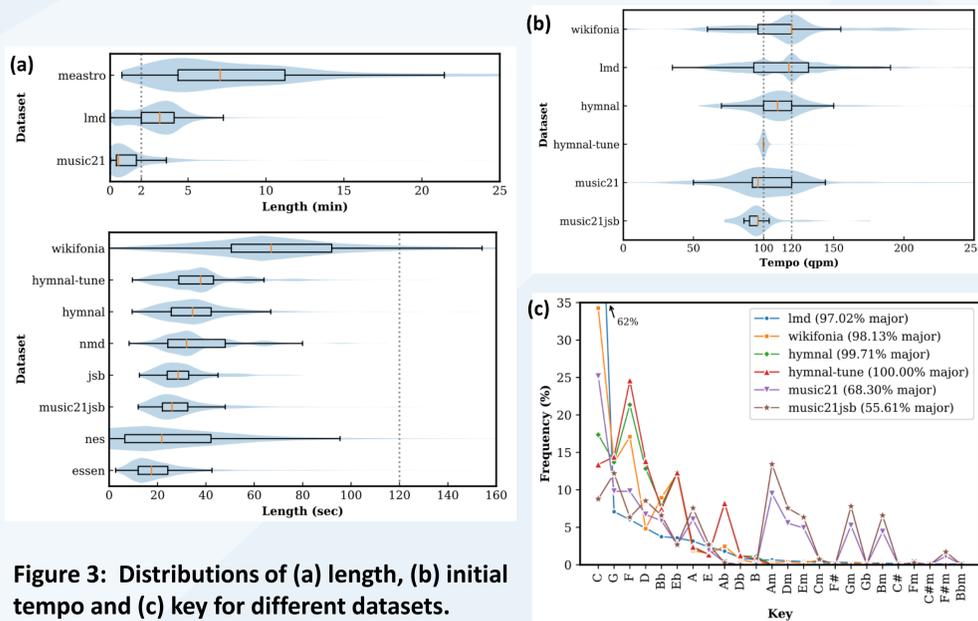


Figure 3: Distributions of (a) length, (b) initial tempo and (c) key for different datasets.

Experiments

- **Representation**—event representation with 128 note-on, 128 note-off, 100 time-shift and 1 end-of-sequence events
- **Data**—64 time steps per sample (4 time steps per quarter note)
- **Training**—predict next event given the previous ones using cross entropy loss
- **Evaluation**—compute the perplexity on 1000 random samples in the test set

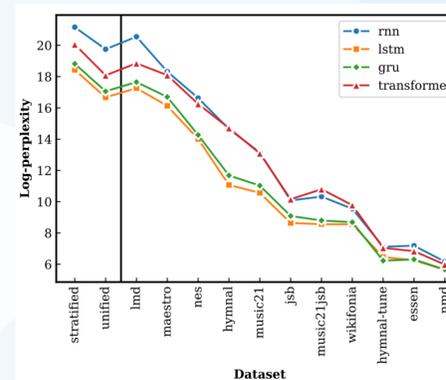


Figure 4: Perplexities for different models on different datasets.

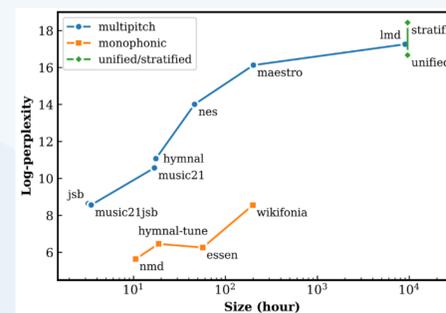


Figure 5: Perplexities for the LSTM model versus dataset size.

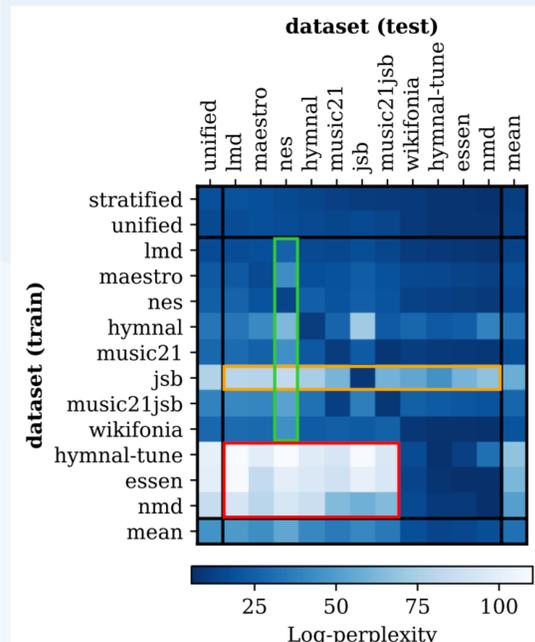


Figure 6: Cross-dataset generalizability results.

Conclusion

- We presented MusPy—a new toolkit that provides essential tools for developing music generation systems.
- We conducted a statistical analysis and experiments on the supported datasets to analyze their relative diversities and cross-dataset generalizabilities.
- We showed that combining heterogeneous datasets could help improve generalizability of a machine learning model.