Andrew McLeod*

James Owers*

Kazuyoshi Yoshii

The MIDI Degradation Toolkit:

Symbolic Music **Augmentation** and Correction

What is it?

A python package to add controlled noise to MIDI excerpts:
https://github.com/JamesOwers/midi_degradation_toolkit

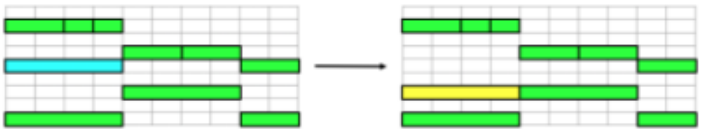
Why?

1. **Train models correcting AMT**
2. **Test model robustness**

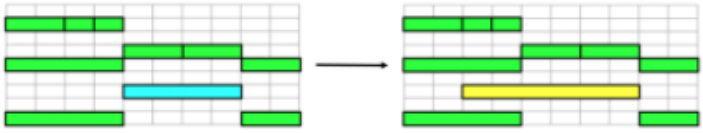
How?

Augment the data with **degradations**

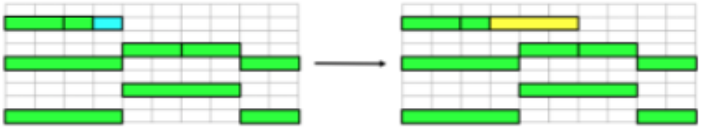
1) Pitch shift



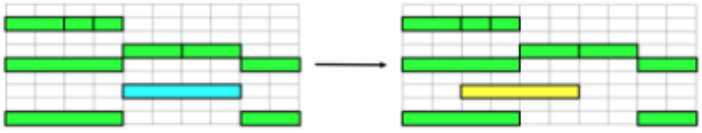
2) Onset shift



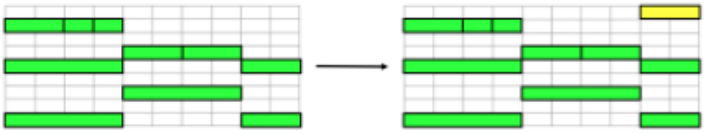
3) Offset shift



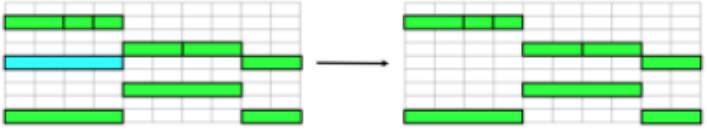
4) Time shift



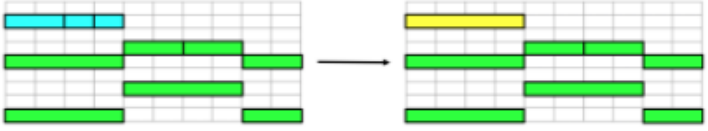
5) Add note



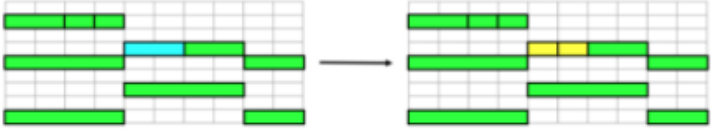
6) Remove note



7) Join notes



8) Split note



Example Workflow

For improving your existing AMT system

Analyse the errors of your AMT system

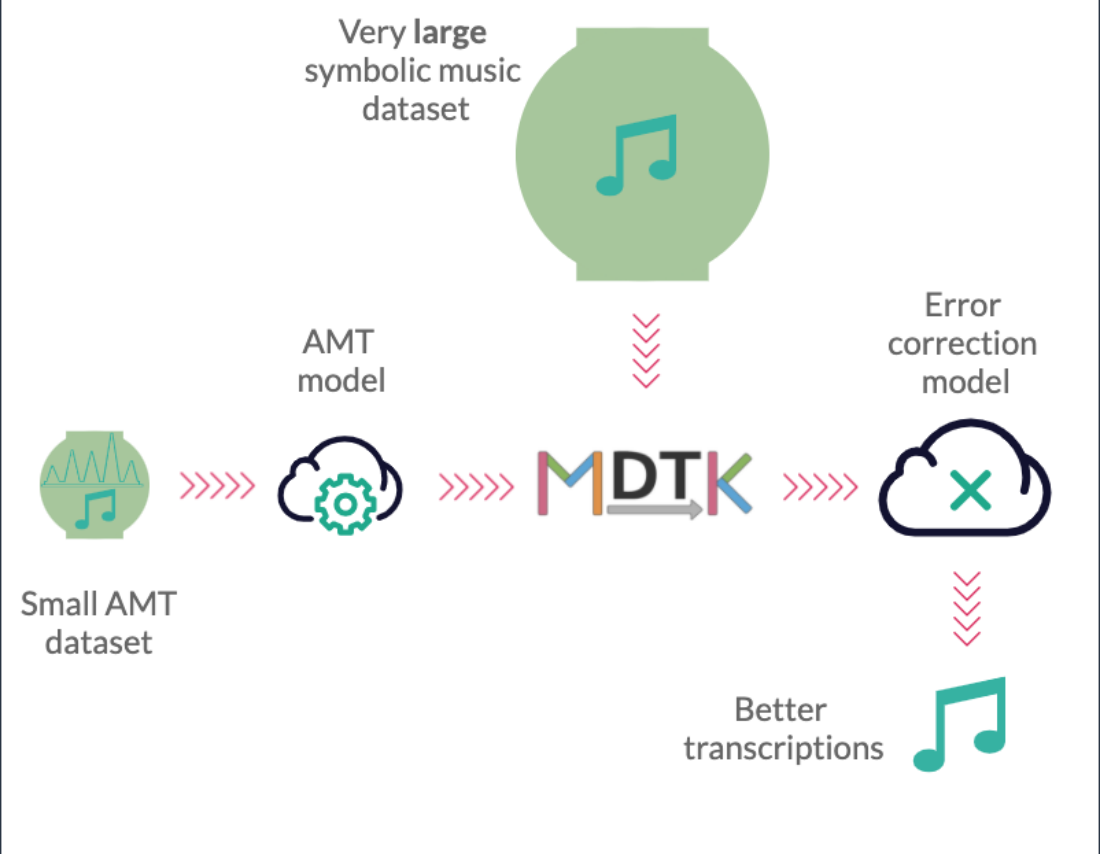
`measure_errors.py`

Create data with the same distribution of errors

`make_dataset.py`

Train a discriminator to correct the errors

`train_task.py`



In summary

MDTK improves AMT Systems by leveraging other data

- Analyses errors of an existing AMT system
- Leverages very large symbolic music datasets
- Creates error data similar to output of AMT system

The user can now train a discriminator on the data provided by MDTK and combine that model with their AMT system.

Other features

In addition to these tools, we also present...

- An example dataset: Altered and Corrupted Midi Excerpts, or ACME dataset
- `Degrader()` class: no need to explicitly create a dataset, can degrade when loading data to the model
- Four proposed tasks of increasing difficulty for training error correcting models, with accompanying baseline models:
  - Error detection - identify if excerpt contains error
  - Error classification - classify the type of error
  - Error location - locate the error within the excerpt
  - Error correction - return a corrected excerpt