## How It Works

### Training

* Runs for **100 epochs**.
* Uses a music generation model that combines:
  + **Temporal Graph Dilated Convolution (TGDC)**
  + **MobileNet Temporal Convolutional Network (MobileNetTCN)**
  + **Sparse Transformer**
  + **Fusion Layer**
* Loss Function: Combined **CrossEntropyLoss (for rhythm)** + **MSELoss (for pitch)**
* Optimizer: **Adam**

### Testing

Computes:

* **ACCURACY**
* **PRECISION**
* **RECALL**
* **F1-SCORE**

### Model Saving

python

torch.save(model.state\_dict(), MODEL\_PATH)

Replace MODEL\_PATH with the trained model name.

### Results Saving

python

df.to\_csv(csv\_path, index=False)

Metrics are saved to a CSV file.

### Dataset

The MIDI dataset should be placed in: dataset/midi

You can also use datasets like the [Lakh MIDI Dataset Clean](https://www.kaggle.com/datasets/imsparsh/lakh-midi-clean)

MUSIC\_GENERATION\

├── \_\_init\_\_.py

├── main.py

├── config.py

├── music\_dataset.py

├── train.py

├── evaluate.py

├── model\_components.py

├── utils.py

├── checkpoints\

│ └── (model files will be saved here)

└── dataset\

├── cache\

└── midi\

### How to Run Training

python

train(MIDI\_PATH)

Replace MIDI\_PATH with the path to your MIDI dataset.

### How to Run Testing

python

run\_eval(MODEL\_PATH, MIDI\_PATH, csv\_path="results.csv")

Replace:

* MODEL\_PATH with the trained model checkpoint.
* MIDI\_PATH with the path to your MIDI dataset.