Music FaderNets: Controllable Music Generation Based On High-Level Features via Low-Level Feature Modelling

Hao Hao Tan, Dorien Herremans

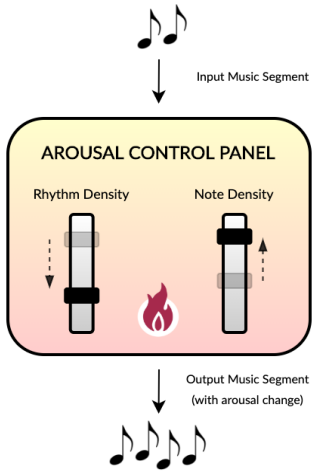
Code : https://github.com/music-fader-nets/
Demo : https://music-fadernets.github.io/

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

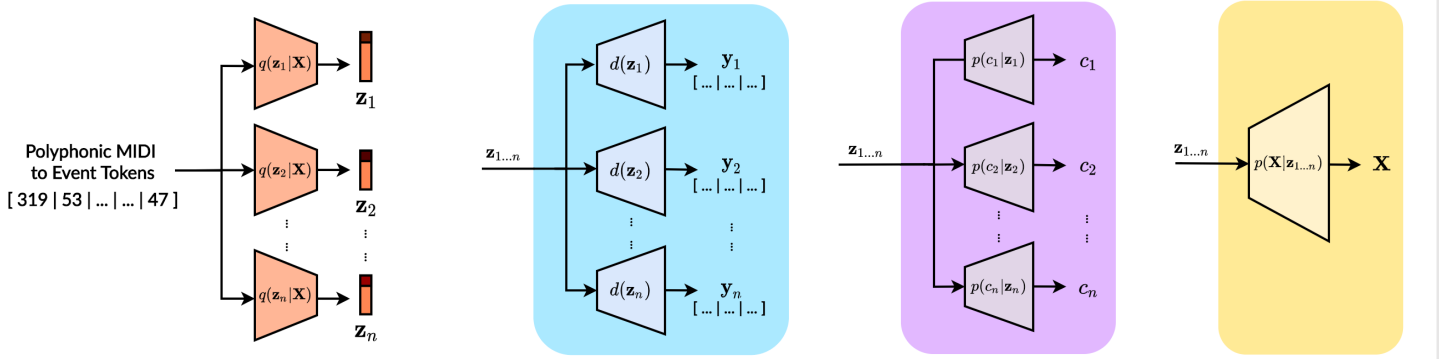
Agency for Science, Technology and Research SINGAPORE

Introduction

- **High-level** musical attributes are **abstract** and often require human annotations.
  However, either labels are **hardly sufficient**, or labels are **too noisy** due to human subjectivity.
- We think that a high-level feature is intrinsically related to a set of **low-level features**.
  HOW they are related can be learnt through data-driven methods, e.g. neural networks.
- We need a **"fader"** like model for generation:
  - Each "knob" controls a low-level feature;
  - A "preset" defines how all the knobs should be moved to exhibit the high-level feature.



Model Formulation



Step 1: Encoding + Latent Regularization
Encode one latent vector for each low-level feature. Map the change of one latent dimension to the change of value in that low-level feature (for the "knob").

Step 2: Discriminator
Enforce the latent vector to contain information of the assigned low-level feature. Trained via supervised learning to reconstruct feature labels.

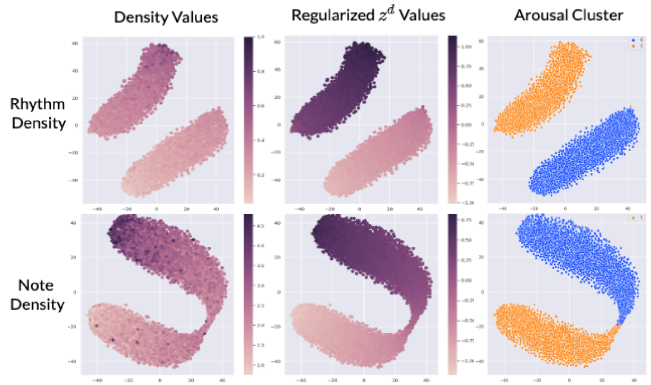
Step 3: Cluster Inference
Inferring high-level feature labels from low-level latent vectors. Trained via semi-supervised learning using GM-VAEs.

Step 4: Reconstruction
Reconstruct the original token sequence from all latent vectors with an autoregressive decoder.

Dataset

- **Piano e-Competition dataset** (~100,000 4-bar sequences) for low-level features:
  - *Rhythm* sequence, e.g.: [onset, hold, hold, offset, onset, ...]
  - *Note* polyphony sequence, e.g. [3, 4, 3, 3, 3, 2, ...]
  - *Rhythm density* = #onsets / length
  - *Note density* = Σ polyphony / length
- **VGMIDI dataset** (~1,000 4-bar sequences) for high-level features (*arousal* values from -1 to 1) annotated by human
  - *High* arousal (*c*=1) if > 0.1
  - *Low* arousal (*c*=0) if < -0.1

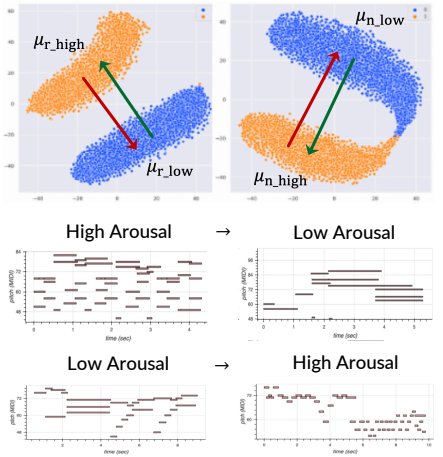
Learnt High-Level Representation



- TSNE shows **clear separation** of high & low arousal mixtures
- **High arousal →** High rhythm density Low note density
- **Low arousal →** Low rhythm density High note density

* The relationship between low-level features and arousal are learnt with only **1%** of arousal labels → works even under **weakly supervised** conditions !

Arousal Style Transfer



- Construct a **"style shifting vector"** from mean point of one Gaussian mixture to another ($\mu_{high} \leftrightarrow \mu_{low}$)
- Style transfer process: $\boldsymbol{X}$ → (encode) → $\boldsymbol{z}$ → (apply shifting vector) → $\boldsymbol{z}'$ → (decode) → $\boldsymbol{X}'$
- **82%** of the subjective test responses agreed with arousal transfer direction.