MusER: Musical Element-Based Regularization for Generating Symbolic Music with Emotion

Shulei Ji, Xinyu Yang*

School of Computer Science and Technology, Xi'an Jiaotong University, Xi'an 710049, P. R. China
taylorji@stu.xjtu.edu.cn, yxyphd@mail.xjtu.edu.cn

Abstract

Generating music with emotion is an important task in automatic music generation, in which emotion is evoked through a variety of musical elements (such as pitch and duration) that change over time and collaborate with each other. However, prior research on deep learning-based emotional music generation has rarely explored the contribution of different musical elements to emotions, let alone the deliberate manipulation of these elements to alter the emotion of music, which is not conducive to fine-grained element-level control over emotions. To address this gap, we present a novel approach employing musical element-based regularization in the latent space to disentangle distinct elements, investigate their roles in distinguishing emotions, and further manipulate elements to alter musical emotions. Specifically, we propose a novel VQ-VAE-based model named MusER. MusER incorporates a regularization loss to enforce the correspondence between the musical element sequences and the specific dimensions of latent variable sequences, providing a new solution for disentangling discrete sequences. Taking advantage of the disentangled latent vectors, a two-level decoding strategy that includes multiple decoders attending to latent vectors with different semantics is devised to better predict the elements. By visualizing latent space, we conclude that MusER yields a disentangled and interpretable latent space and gain insights into the contribution of distinct elements to the emotional dimensions (i.e., arousal and valence). Experimental results demonstrate that MusER outperforms the state-of-the-art models for generating emotional music in both objective and subjective evaluation. Besides, we rearrange music through element transfer and attempt to alter the emotion of music by transferring emotion-distinguishable elements.

Introduction

Generating music with emotion is a crucial and challenging research problem in the field of automatic music generation (Ji, Yang, and Luo 2023). The generated emotional music can convey and elicit human emotions, thereby enhancing human-machine interaction. The application scenarios for generating music that evokes specific emotions are gradually increasing, e.g., composing soundtracks (Ferreira, Lelis, and Whitehead 2020) for media forms such as user-generated video content, video games, and movies. Furthermore, generating emotional music holds promise for medical applications (Pratt, Abel, and Skidmore 1995), such as music therapy. For annotating emotions, the two-dimensional arousal-valence (A-V) emotion model (Russell 1980) is widely adopted in the literature. Within this model, the two dimensions, namely arousal and valence, respectively indicate the level of autonomic activation and pleasantness. These two dimensions divide the A-V model into four quadrants, each corresponding to a specific class of emotions. In this paper, we employ the four quadrants (4Q) as the emotion labels for generating emotional music.

Music conveys emotion influenced by multiple musical elements, such as pitch, duration, velocity, and tempo. However, existing research (Ferreira, Lelis, and Whitehead 2020; Hung et al. 2021; Ferreira et al. 2022; Neves, Fornari, and Florindo 2022; Bao and Sun 2022; Wang, Zhang, and Zhou 2023) on deep learning-based emotional music generation has not thoroughly explored the relationship between individual musical elements and emotion, nor have they investigated how manipulations of musical elements impact emotional expressions. Instead, with the help of music datasets with emotional labels, end-to-end models were often employed to learn the distributions of musical emotion features.

To investigate the relationships between distinct musical elements and emotions, it is first necessary to disentangle individual elements within the latent space. For a certain element, its variation pattern over time seems to be particularly important for conveying emotions as opposed to its statistical properties. For instance, the same pitch distribution (implying the same pitch range, number of pitch classes, etc.) may produce either positive or negative valence, depending on the shifts in pitch over time. Therefore, two requirements should be satisfied for the disentangled latent representation of a musical element: i) it should encode a certain element independent of other elements; ii) it should learn not only the statistical properties of the element but also its variation patterns over time.

Previous disentanglement learning techniques for music (Yang et al. 2019; Chen et al. 2020a; Chen, Xia, and Dubnov 2020b; Wei and Xia 2021) have learned to disentangle musical elements, but they were limited to decoupling only a few elements (i.e., pitch and rhythm). Moreover, these approaches lacked explicit modeling of the variations within

---

*Corresponding author.

musical element sequences, which is crucial for learning musical emotions. Existing latent space regularization techniques (Hadjeres, Nielsen, and Pachet 2017; Pati and Lerch 2021) have effectively established monotonic relationships between the music attribute and latent variable, where the continuous attribute value is encoded along a particular dimension of the latent variable. However, these techniques cannot be directly applied to the musical element due to its discrete nature and sequential characteristic.

To address the above issues, we propose a novel musical element disentanglement module (MED) with a regularization loss acting on the latent space. Specifically, we extend a single latent variable into a sequence of latent variables and force its specified dimensions to encode the individual discrete element sequence. To obtain the latent variable sequence, we employ the vector-quantized variational autoencoder (VQ-VAE) (Van Den Oord, Vinyals et al. 2017) and map its learned discrete latent codes into a sequence of latent vectors. Building upon the disentangled latent space, we further propose a new two-level decoding module (TD) that incorporates multiple decoders. Different decoders selectively attend to latent variable sequences characterized by diverse semantics and consequently improve the prediction of corresponding elements.

Our proposed model, referred to as MusER (**Mus**ical **E**lement-based **R**egularization), utilizes MED and TD for musical element disentanglement to investigate the relationships between musical elements and emotions. Post-training, the visual analysis of the latent space revealed successful disentanglement for different elements. Notably, certain elements (e.g., velocity) exhibited strong emotion-discriminative characteristics. Experiments showed that MusER outperformed prior methods for generating emotional music in both objective music metrics and emotional expression. Moreover, element transfer was achieved by exchanging the element-specific latent variables, and musical emotion could be altered by transferring the emotion-distinguishable elements. We provide the generated music examples and code at a GitHub repo: https://github.com/Tayjsl97/MusER.

Related Work

Emotion-Conditioned Music Generation

In previous studies, emotions have been incorporated into models either as external conditions (Makris, Agres, and Herremans 2021; Grekow 2021; Neves, Fornari, and Florindo 2022; Ji and Yang 2023) or as part (e.g., prefix token) of the music representation (Hung et al. 2021; Wang, Zhang, and Zhou 2023) to guide the generation of emotional music, which is also known as conditional sampling (Ferreira et al. 2022). Search methods such as beam search (Ferreira, Lelis, and Whitehead 2020; Bao and Sun 2022) and Monte Carlo tree search (Ferreira et al. 2022) have also been utilized to steer the music sequence toward a specific emotion. However, prior studies have neglected to explore the relationships between emotions and musical elements, as well as the potential of manipulating musical elements to alter musical emotions. Although Hung et al. (2021) revealed the distribution disparities of some musical elements under distinct emotions, they did not employ these findings to guide emotional music generation. In this paper, we aim to disentangle the latent space of different musical elements, unveil their roles in distinguishing emotions, and manipulate emotion-distinguishable elements to induce alterations in musical emotion.

Interpretable Latent Representation Learning

Most studies on controllable music generation have embraced VAE-based or VAE-inspired architectures to learn interpretable latent representations, which can increase model transparency and facilitate control over the generated music.

Disentangling Latent Space. (Akama 2019) introduced a model that leveraged domain knowledge to decouple the latent space, with a specific emphasis on three musical concepts: rhythm, contour, and fragmentation & consolidation. (Yang et al. 2019; Chen, Xia, and Dubnov 2020b; Wei and Xia 2021) disentangled the pitch and rhythm representations using EC^2-VAE with a rhythm decoder and a global decoder. In contrast, (Chen et al. 2020a) factorized pitch and rhythm by introducing two encoders that consider pitch and rhythm information separately. Additionally, (Wang et al. 2020) focused on chord and texture disentanglement, and (Luo et al. 2020) aimed to disentangle style and content.

Regularizing Latent Space. (Hadjeres, Nielsen, and Pachet 2017) proposed a geodesic latent space regularization for VAE to bind a displacement in some directions of the latent space to a qualitative change of the attributes. (Pati and Lerch 2019, 2021) proposed an attribute regularization loss to enforce a monotonic relationship between the attribute value and the specific dimension of latent variable. Both methods focus on encoding continuous-valued attributes. Unlike the prior studies, regularizing latent space to disentangle musical elements remains an unexplored area of research. Inspired by the work of (Pati and Lerch 2021), the regularization method proposed in this paper extends the continuous attributes to the discrete sequences of musical elements.

Methodology

In this section, we introduce our proposed MusER for musical element disentanglement and emotional music generation. Figure 1 shows the model architecture of MusER, which is based on VQ-VAE and consists of four key components: i) an encoder that encodes the symbolized music representation into the latent space; ii) the vector quantization process that maps the encoder output into the matched latent variables through a nearest-neighbor lookup; iii) a musical element disentanglement module (MED) that disentangle musical elements through regularizing the matched latent variables; iv) a two-level decoding module (TD) that reconstructs music from the matched latent variables.

Background

Music Representation. A preliminary step in symbolic music generation with neural sequence models is to symbolize the music into a discrete representation. To represent music, we adopt Compound Word (CP) (Ren et al.

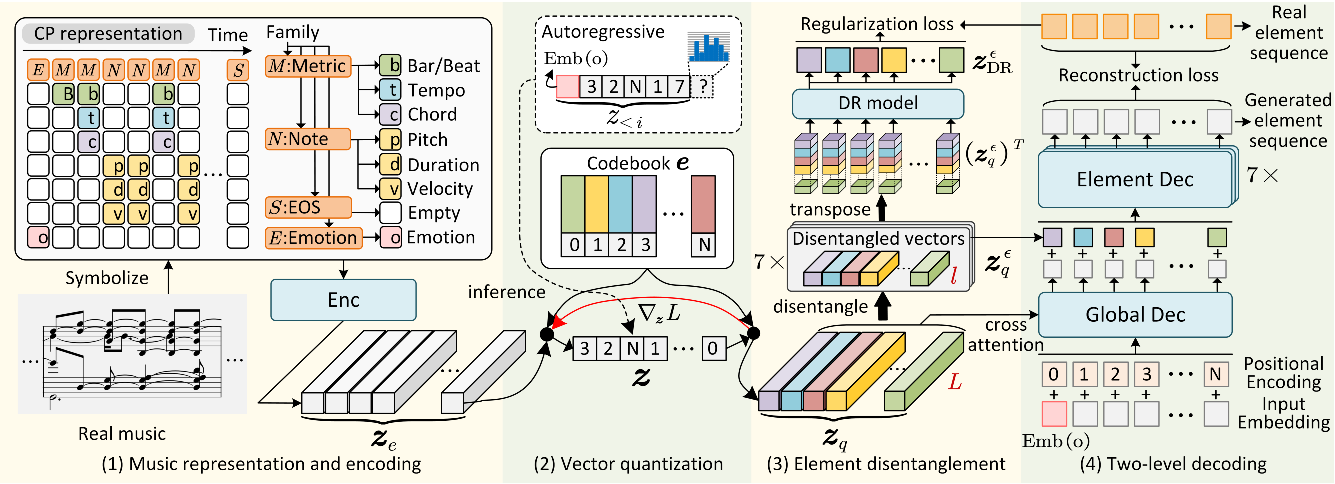


Figure 1: The architecture of MusER consisting of four components: music representation and encoding, vector quantization, musical element disentanglement (MED), and two-level decoding (TD). DR is the acronym for dimensionality reduction. Emb(o) denotes the emotion embedding. The gradient $\nabla_{\boldsymbol{z}} L$ (in red) is passed unaltered to the encoder during the backwards pass. The dashed box indicates that a conditional autoregressive model is trained to predict discrete codes during inference.

2020) as it allows the simultaneous occurrence of multiple musical events to alleviate the issue of long sequences. Following (Hung et al. 2021), the CPs are partitioned into four families: 3-*note*, 2-*metric*, 1-*emotion*, and 0-*end-of-sequence (EOS)*, as illustrated in Figure 1. The sequence of CPs is formulated as $S_{CP} = \{cp_i\}_{i=1}^{N}$, where $cp_i = \{x_i^{\mathrm{f}}, x_i^{\mathrm{b}}, x_i^{\mathrm{t}}, x_i^{\mathrm{c}}, x_i^{\mathrm{p}}, x_i^{\mathrm{d}}, x_i^{\mathrm{v}}, x_i^{\mathrm{o}}\}$, N is the sequence length and i indicates the ith time step. The symbol set $\mathcal{E} = \{\mathrm{f}, \mathrm{b}, \mathrm{t}, \mathrm{c}, \mathrm{p}, \mathrm{d}, \mathrm{v}, \mathrm{o}\}$ correspond to eight types of event tokens: family, bar-beat, tempo, chord, pitch, duration, velocity, and emotion. Note that the missing token types per time step are filled with "empty" tokens, ensuring a consistent prediction of 8 tokens at each step. The sequence of element ϵ is formulated as $\boldsymbol{x}^{\epsilon} = \{x_i^{\epsilon}\}_{i=1}^{N}$, where $\epsilon \in \mathcal{E}$. Thus, the sequence of CPs can be transformed as $S_{CP} = \{\boldsymbol{x}^{\epsilon} | \epsilon \in \mathcal{E}\}$. In this paper, we aim to disentangle seven types of musical elements excluding emotion (i.e., o), denoted as $\hat{\mathcal{E}} = \mathcal{E} \setminus \mathrm{o}$.

VQ-VAE. As depicted in Figure 1, the encoder output $\boldsymbol{z}_e$ goes through a nearest-neighbor lookup to match one of the embedding vectors within the codebook $\boldsymbol{e} = [e_1, ..., e_K]^T \in \mathbb{R}^{K \times L}$, where K is the size of the discrete latent space (i.e., a K-way categorical), and L is the dimensionality of each embedding vector. The indices of the lookup are saved as the discrete latent codes $\boldsymbol{z}$. The matched vector $\boldsymbol{z}_q$ is the latent vector sequence to be disentangled and the input for the decoder to reconstruct the element sequences $\boldsymbol{x}$.

$$\begin{aligned} &\boldsymbol{x} = \mathrm{Dec}(\boldsymbol{z}_q),\ \boldsymbol{z}_q = \left[z_q^1, z_q^2, ..., z_q^N\right]^T, \\ &z_q^i = e_k,\ \text{where } k = \mathrm{argmin}_j \left\| z_e^i - e_j \right\|_2 \qquad (1) \\ &\boldsymbol{z}_e = \left[z_e^1, z_e^2, ..., z_e^N\right]^T,\ \boldsymbol{z}_e = \mathrm{Enc}(\boldsymbol{x}) \end{aligned}$$

N is the length of the latent vector sequence, which is equivalent to the input sequence length. e_j is the jth embedding in the codebook. z_e^i and z_q^i are the entries at the ith step in sequences of $\boldsymbol{z}_e$ and $\boldsymbol{z}_q$. Enc and Dec denote encoder and decoder, respectively.

Musical Element-Based Regularization

We formulate a novel regularization loss acting on the latent space to disentangle different musical elements. Diverging from prior approaches that encode continuous attributes utilizing specific dimensions of the latent variable (Hadjeres, Nielsen, and Pachet 2017; Pati and Lerch 2021), our model encodes discrete sequences of musical elements along the specific dimensions of the latent vector sequences, i.e.,

$$\boldsymbol{z}_q = \bigoplus\nolimits_{\epsilon} \boldsymbol{z}_q^{\epsilon},\ \epsilon \in \hat{\mathcal{E}} \tag{2}$$

where $\boldsymbol{z}_q \in \mathbb{R}^{N \times L}$ is the sequence of L-dimensional (D) latent vectors to be regularized, and $\boldsymbol{z}_q^{\epsilon} \in \mathbb{R}^{N \times l}$ is the l-D latent vector sequence used for encoding the element ϵ, where $L = l \times 7$ and $l = 16$ is adopted in this paper.

To achieve the regularization, we establish a correspondence between variations in musical element sequences and the corresponding variations in latent variable sequences. Figure 2 shows the schematic illustration of the regularization method. Mathematically, let $\boldsymbol{x}_i^{\epsilon},\ \boldsymbol{x}_j^{\epsilon} \in \mathbb{R}^{N \times 1}$ denote two input sequences of element ϵ, generated using the corresponding sequences of latent vectors $\boldsymbol{z}_{q_i}^{\epsilon},\ \boldsymbol{z}_{q_j}^{\epsilon} \in \mathbb{R}^{N \times l}$. At the t-th time step, if $x_{t,i}^{\epsilon} > x_{t,j}^{\epsilon}$ for any i and j, then it should hold that $z_{\mathrm{DR}_{t,i}}^{\epsilon} > z_{\mathrm{DR}_{t,j}}^{\epsilon}$, where $x_{t,i}^{\epsilon}$ and $x_{t,j}^{\epsilon}$ represent the 1-D element inputs, $z_{\mathrm{DR}_{t,i}}^{\epsilon}$ and $z_{\mathrm{DR}_{t,j}}^{\epsilon}$ represent the 1-D latent vectors derived from the l-D latent vectors $z_{q_{t,i}}^{\epsilon}$ and $z_{q_{t,j}}^{\epsilon}$, which will be elaborated on in the follow-up.

Specifically, the regularization loss is formulated as:

$$L_R = \sum\nolimits_{\epsilon} \mathrm{MAE}(\tanh(\boldsymbol{M}^{\epsilon,R}) - \mathrm{sgn}(\boldsymbol{M}^{\epsilon})),\ \epsilon \in \hat{\mathcal{E}} \tag{3}$$

where $\boldsymbol{M}^{\epsilon}$ and $\boldsymbol{M}^{\epsilon,R}$ are two distance matrices that represent the variations in musical element sequences and latent

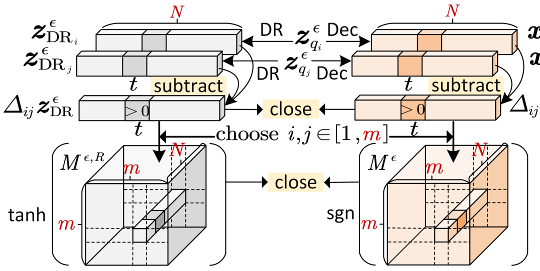


Figure 2: The schematic illustration of the regularization for element ϵ. DR represents dimensionality reduction. Dec denotes the decoding module. "subtract" means the subtraction of entries at the same position in two sequences, and Δ denotes the result of the subtraction. "close" indicates that two vectors or matrices are expected to be as close as possible.

variable sequences, respectively. $\mathrm{MAE}(\cdot)$ is the mean absolute error, $\tanh(\cdot)$ is the hyperbolic tangent function, and $\mathrm{sgn}(\cdot)$ is the sign function. $\mathrm{sgn}(\cdot)$ is used to obtain the signs of the differences between two arbitrary element sequences, ignoring the magnitudes of the differences. $\tanh(\cdot)$ is utilized to normalize the range of values in $\boldsymbol{M}^{\epsilon,R}$ to $[-1,1]$, keeping the same range as $\mathrm{sgn}(\boldsymbol{M}^{\epsilon})$ (Pati and Lerch 2021). For musical element ϵ, $\mathrm{MAE}(\cdot)$ forces the sequences of element-specific latent vectors to learn the corresponding sequences of musical elements.

When considering a mini-batch of m samples, $\boldsymbol{M}^{\epsilon}$, $\boldsymbol{M}^{\epsilon,R} \in \mathbb{R}^{m\times m\times N}$ are calculated as follows:

$$\boldsymbol{M}^{\epsilon} = [M^{\epsilon}_1, ..., M^{\epsilon}_N],\ M^{\epsilon}_t(i,j) = x^{\epsilon}_{t,i} - x^{\epsilon}_{t,j}$$
$$\boldsymbol{M}^{\epsilon,R} = [M^{\epsilon,R}_1, ..., M^{\epsilon,R}_N],\ M^{\epsilon,R}_t(i,j) = z^{\epsilon}_{\mathrm{DR}_{t,i}} - z^{\epsilon}_{\mathrm{DR}_{t,j}} \tag{4}$$

where $t \in [1,N]$ and $i,j \in [1,m]$. To obtain $\boldsymbol{z}^{\epsilon}_{\mathrm{DR}} \in \mathbb{R}^{m\times N\times 1}$ from $\boldsymbol{z}^{\epsilon}_q \in \mathbb{R}^{m\times N\times l}$, we adopted a transformer-based dimensionality reduction (DR) model, as shown in the MED module in the Figure 1. Specifically, we first transpose the latent vector $\boldsymbol{z}^{\epsilon}_q \in \mathbb{R}^{m\times N\times l}$ into $(\boldsymbol{z}^{\epsilon}_q)^T \in \mathbb{R}^{m\times l\times N}$. Next, we utilize the DR model to automatically aggregate the l-D vector into a single dimension. The output at the first step of the DR model, $\boldsymbol{z}^{\epsilon}_{\mathrm{DR}} \in \mathbb{R}^{m\times N\times 1}$, is utilized to calculate $\boldsymbol{M}^{\epsilon,R}$. We use the transformer as the DR model since its self-attention mechanism can perform a weighted sum over l dimensions to produce an optimal dimension adaptively.

Two-level Decoding

Drawing inspiration from the Compound Word (CP) Transformer (Hsiao et al. 2021), wherein different feed-forward heads are employed for predicting tokens of distinct types, we devise a two-level decoding module, as depicted in Figure 1. The purpose of this decoder module is to predict different elements in a hierarchical manner, from coarse to fine. Each decoder receives guidance from its corresponding latent vectors, aligned with its prediction objectives.

Concretely, the decoding module comprises a global decoder $\mathrm{Dec}_{\mathrm{G}}$ and seven element decoders Dec_{ϵ}. The $\mathrm{Dec}_{\mathrm{G}}$ captures the global information of the music and attends to the whole latent vectors $\boldsymbol{z}_q$ through the cross-attention module of the transformer. Its output is summed with each l-D sub-latent vectors $\boldsymbol{z}^{\epsilon}_q$ and then sent to their respective element decoders for reconstructing the element sequences, i.e.,

$$\boldsymbol{h}^{\epsilon} = \mathrm{Dec}_{\epsilon}(\boldsymbol{h} + \mathrm{Linear}(\boldsymbol{z}^{\epsilon}_q)),\ \boldsymbol{h} = \mathrm{Dec}_{\mathrm{G}}(\boldsymbol{x}, \boldsymbol{z}_q) \tag{5}$$

where $\boldsymbol{h}$ and $\boldsymbol{h}^{\epsilon}$ denote the outputs of $\mathrm{Dec}_{\mathrm{G}}$ and Dec_{ϵ}. $\mathrm{Linear}(\cdot)$ represents a linear layer to align the dimension of $\boldsymbol{z}^{\epsilon}_q$ with $\boldsymbol{h}$. Following (Hsiao et al. 2021), a two-stage prediction setting is adopted wherein we predict the [family] token first and then predict the remaining elements given the [family] token. The family token can explicitly specify the token group to be predicted at each step.

Training and Inference

Training Objective. The loss function L in our proposed model includes: i) a reconstruction loss for predicting musical elements, ii) a codebook loss for updating the codebook, iii) a commitment loss to ensure the encoder commits to an embedding and its output does not grow (Van Den Oord, Vinyals et al. 2017), and iv) a regularization loss (i.e., Equation (3)) for disentangling different musical elements.

$$L = \underbrace{\log p(\boldsymbol{x}|\boldsymbol{z}_q)}_{\text{reconstruction}} + \underbrace{\|\mathrm{sg}\,[\boldsymbol{z}_e] - \boldsymbol{e}\|^2_2}_{\text{codebook}} + \beta\underbrace{\|\boldsymbol{z}_e - \mathrm{sg}\,[\boldsymbol{e}]\|^2_2}_{\text{commitment}} + \alpha L_R \tag{6}$$

where $\mathrm{sg}[\cdot]$ represents the stop gradient operator, α and β are the loss weights. Note that in this paper the reconstruction loss is computed using the cross-entropy loss, and the L_2 codebook loss is replaced with exponential moving averages (EMA) since EMA tends to converge faster than the L_2 loss.

Moreover, before inference, a prior model needs to be trained to learn the categorical distribution over the discrete code $\boldsymbol{z}$. As shown in the dashed box in Figure 1, we fit a conditional autoregressive distribution over $\boldsymbol{z}$:

$$p(\boldsymbol{z}) = \prod\nolimits_i p(z_i|\{z_j|j<i\}, \mathrm{Emb}(\mathrm{o})) \tag{7}$$

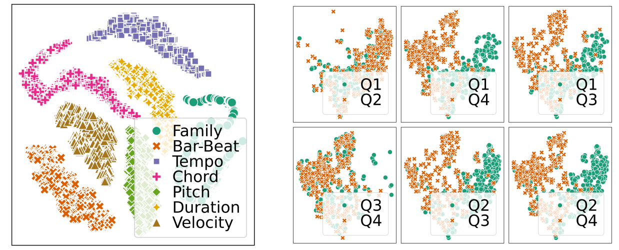
where $\mathrm{Emb}(\mathrm{o})$ is the emotion embedding as a condition. The cross-entropy loss is adopted to train this prior model.

Inference. During inference, we first utilize the prior model to generate discrete latent codes $\boldsymbol{z}$ and obtain the sequence of latent variables $\boldsymbol{z}_q$ by looking up the codebook. Subsequently, $\boldsymbol{z}_q$ and disentangled $\boldsymbol{z}^{\epsilon}_q$ are respectively fed to $\mathrm{Dec}_{\mathrm{G}}$ and Dec_{ϵ} for generating music autoregressively.

Additionally, MusER can rearrange music $\boldsymbol{x}_A$ by replacing $\boldsymbol{z}^{\epsilon}_{q_A}$ for element ϵ with $\boldsymbol{z}^{\epsilon}_{q_B}$ derived from a reference piece $\boldsymbol{x}_B$ and creating new hybrid music $\boldsymbol{x}_{AB}$. This process is called *element transfer* in this paper. Note that the prerequisite for performing element transfer is that the element latent spaces are successfully disentangled, as demonstrated in the next subsection. Taking velocity (i.e., v) transfer as an example, the process is as follows:

$$\begin{aligned} \boldsymbol{x}_{AB} &= \mathrm{Dec}(\boldsymbol{z}_{q_{AB}}) \\ \boldsymbol{z}_{q_{AB}} &= \Big(\bigoplus\nolimits_{\epsilon} \boldsymbol{z}^{\epsilon}_{q_A}\Big) \oplus \boldsymbol{z}^{\mathrm{v}}_{q_B},\ \epsilon \in \hat{\mathcal{E}} \setminus \mathrm{v} \\ \bigoplus\nolimits_{\epsilon} \boldsymbol{z}^{\epsilon}_{q_k} &= \mathrm{Enc}(\boldsymbol{x}_k),\ k \in \{A,B\},\ \epsilon \in \hat{\mathcal{E}} \end{aligned} \tag{8}$$

where $\oplus$ and $\bigoplus_{\epsilon}$ denote vector concatenation and vector concatenation over ϵ, respectively.



(a) The whole latent space w.r.t. the element types.

(b) The velocity latent space w.r.t. four emotion quadrants.

Figure 3: t-SNE visualization of latent space.

	Valence (V)		Arousal (A)		V & A	
	Q1-Q2	Q3-Q4	Q1-Q4	Q2-Q3	Q1-Q3	Q2-Q4
f	0.071	0.004	0.070	0.175	0.103	0.210
t	0.063	0.014	0.004	0.100	0.033	0.147
c	0.048	0.011	0.006	0.083	0.028	0.116
b	**0.083**	0.015	0.078	0.222	0.110	0.270
p	0.054	**0.032**	0.148	0.260	0.239	0.349
d	0.050	0.004	0.146	0.249	0.144	0.252
v	0.080	0.017	**0.222**	**0.341**	**0.286**	**0.404**

Table 1: The silhouette coefficient (SC) scores for evaluating the clustering of element latent spaces regarding the four emotion quadrants. $\mathrm{f}, \mathrm{t}, \mathrm{c}, \mathrm{b}, \mathrm{p}, \mathrm{d}, \mathrm{v}$ correspond to family, tempo, chord, bar-beat, pitch, duration, and velocity, respectively. In terms of distinguishing between two quadrants using distinct elements, the largest SC score is marked in bold and the second largest is underlined.

Latent Space Visualization

We employ t-SNE to visualize the latent space. Concretely, we first visualize the L-D latent space regarding the element types, as depicted in Figure 3(a), demonstrating successful element disentanglement. Then, we visualize the l-D element latent space with respect to 4Q. Figure 3(b) presents an example visualization of velocity latent space. Lastly, the Silhouette Coefficient (SC) (Rousseeuw 1987) is utilized to evaluate the emotional clustering within the element-specific latent space, as shown in Table 1. SC ranges from -1 and 1, with larger values indicating more coherent clusters. We observed that:

(1) Distinguishing V (i.e., Q1 vs. Q2 or Q3 vs. Q4) based on individual musical elements is challenging since the best-performing element in distinguishing V attains an SC score that remains proximate to 0. This observation implies the ongoing difficulty in quantitatively interpreting the relationship between human perception of V and specific musical elements.

(2) As for distinguishing A (i.e., Q1 vs. Q4 or Q2 vs. Q3), some elements exhibit much stronger proficiency, where velocity yields the best results followed by pitch. This suggests the possibility of influencing A by manipulating the A-distinguishable elements. Intriguingly, despite the indistinguishable pitch distributions among 4Q, the pitch latent space can discriminate A, indicating the latent space appears to encode not only statistic properties but also other aspects like variation patterns.

(3) Furthermore, it is intuitive that all elements perform better in discerning emotions when both A and V are different (i.e., Q1 vs. Q3 or Q2 vs. Q4), as opposed to situations where only one dimension (A or V) differs.

Experiments

Experiment Settings

Dataset. Two datasets are used in this paper to train the proposed model. The first one is the **Pop1k7** dataset[1] (Hsiao et al. 2021), which contains 1748 piano covers of pop songs automatically transcribed by a piano transcription model (Hawthorne et al. 2018) and converted into MIDI files. The second one is the **EMOPIA** dataset[2] (Hung et al. 2021), a multi-modal database focusing on perceived emotion in pop piano music. This dataset comprises 1087 music clips from 387 piano solo performances and clip-level emotion labels annotated by dedicated annotators. The emotion labels are the 4Q in the circumplex model of affect (Russell 1980).

Implementation Details. All modules in MusER, comprising the encoder, decoders, DR model, and categorical prior model, take linear transformer (Katharopoulos et al. 2020) as backbone due to its lightweight and linear complexity in attention calculation. Following (Hung et al. 2021), we set the length of the token sequence to 1024 for both datasets and apply specific sampling policies (temperature sampling and nucleus sampling (Holtzman et al. 2020)) for different elements. The implementation details of CP and sampling policies almost coincide with (Hsiao et al. 2021).

Compared Models. We compare MusER with two previous models for emotional music generation on the EMOPIA dataset. The first one is the CP Transformer variant, initially introduced by (Hung et al. 2021) as a prototype for symbolic emotional music generation on the EMOPIA dataset. The second one is the Transformer GAN (Neves, Fornari, and Florindo 2022). Both of these models were pre-trained on Pop1k7 and then fine-tuned on EMOPIA. To be consistent with prior work for a fair comparison, all models in this paper adhere to this training pipeline.

Objective Metrics. In line with the compared models (Hung et al. 2021; Neves, Fornari, and Florindo 2022), we use **Pitch Range (PR)**, **Number of Pitch Classes (NPC)**, and **Polyphony (POLY)** as metrics to objectively evaluate the generated music with the real music. These metrics are calculated using the Muspy library (Dong et al. 2020). In addition to these piece-level metrics, we computed PR, NPC, and POLY at the bar level, i.e., **B-PR**, **B-NPC**, and **B-POLY**, for finer evaluation. Aligning with the setting used for computing objective metrics in (Hung et al. 2021; Neves, Fornari, and Florindo 2022), we generated 400 samples (100 for each emotion quadrant) for each model and calculate the average of the metrics values as the final scores.

---

[1] https://github.com/YatingMusic/compound-word-transformer

[2] https://annahung31.github.io/EMOPIA/

Model	PR	p-value	B-PR	p-value	NPC	p-value	B-NPC	p-value	POLY	p-value	B-POLY	p-value
EMOPIA (Real data)	50.78	-	32.57	-	8.48	-	4.61	-	5.90	-	2.09	-
CP Transformer *	49.52	0.008	27.09	0.012	8.52	**0.255**	4.00	0.005	4.30	**0.069**	1.79	0.010
Transformer GAN **	**50.73**	-	-	-	9.45	-	-	-	4.43	-	-	-
MusER (Trans_CA_Dec$_{\mathrm{G}+\epsilon}$)	53.47	0.005	**33.02**	0.040	8.57	**0.494**	4.43	**0.070**	4.63	**0.421**	1.94	8.9e-5
w/ distinct configuration												
Mean_CA_Dec$_{\mathrm{G}+\epsilon}$	50.58	**0.210**	30.45	0.002	**8.49**	**0.132**	4.38	0.044	4.25	0.038	1.94	2.8e-6
Trans_Concat_Dec$_{\mathrm{G}+\epsilon}$	54.64	6.9e-6	33.18	0.003	8.99	4.0e-4	4.49	**0.075**	4.75	**0.733**	**2.10**	0.002
Mean_Concat_Dec$_{\mathrm{G}+\epsilon}$	57.43	7.5e-5	34.15	0.020	8.87	0.040	4.38	0.001	4.64	**0.640**	1.92	4.0e-4
w/o TD												
Trans_CA_Dec$_{\mathrm{G}}$	55.37	**0.112**	33.19	0.004	9.58	1.0e-4	**4.67**	0.041	4.39	**0.139**	2.07	2.0e-4
Trans_None_Dec$_{\epsilon}$	53.46	0.003	33.20	5.0e-4	9.47	0.001	4.94	0.034	**5.15**	**0.767**	2.03	3.0e-4
w/o MED												
None_CA_Dec$_{\mathrm{G}}$	49.26	0.005	29.15	2.0e-4	8.84	**0.082**	4.25	5.0e-4	4.63	**0.096**	1.84	7.0e-6
None_Concat_Dec$_{\mathrm{G}}$	52.41	**0.175**	31.31	0.007	9.04	**0.064**	4.45	0.015	4.70	**0.339**	1.94	6.4e-8
None_Concat_Dec$_{\mathrm{G}}$ (VAE)	50.86	**0.392**	30.78	1.4e-6	8.72	**0.449**	4.13	0.007	4.50	**0.285**	1.82	2.0e-4

* The results are calculated based on 400 regenerated music samples using the trained model checkpoint[2] provided by (Hung et al. 2021).
** The results are directly derived from (Neves, Fornari, and Florindo 2022).

Table 2: The results of objective evaluation. The top block presents the comparison with previous models and the bottom block shows the model configuration comparison. The best score is highlighted in bold and the second best is underlined. All p-values greater than 0.05 are marked in bold. For a specific model configuration, the underlines divide the model name into three parts, implying the way of dimensionality reduction (DR), the way of feeding z_q into the global decoder, and the adopted decoders.

It is noteworthy that the generated results are generally better when the metric scores are closer to those of the real data (EMOPIA). Additionally, for statistical analysis, a two-tailed t-test is employed to compare the objective scores of the models with those of the real data. A p-value below 0.05 is considered statistically different, and the objective results of the models are expected to not statistically deviate from those of the real data, i.e., $p \geqslant 0.05$.

Objective Evaluation

Model Configuration Study. We conducted a series of experiments to investigate the impact of different model configurations on the results, as presented in the bottom block in Table 2. Specifically, we compared the transformer-based (Trans) DR with dimension averaging (Mean) and explored two approaches for feeding z_q into the decoder, namely cross-attention (CA) and concatenation with the input embeddings (Concat). We conducted ablation studies on TD (i.e., adopting only $\mathrm{Dec}_{\mathrm{G}}$ or Dec_{ϵ}) as well as MED. Note that models w/o MED automatically contain only $\mathrm{Dec}_{\mathrm{G}}$ (i.e., w/o TD) due to the absence of disentangled latent vectors. Moreover, we compared the discrete and continuous latent spaces (i.e., VQ-VAE vs. VAE) of models w/o MED when adopting Concat to feed z_q into the decoder, as shown in the last two rows in Table 2.

When considering the p-value, the best performance is achieved by MusER (i.e., Trans_CA_Dec$_{\mathrm{G}+\epsilon}$) that combines transformer-based DR and two-level decoders while sending z_q into the decoder via CA, though it performs poorly in PR and B-POLY. When focusing solely on the DR approach, Trans outperforms Mean. Between the two ways of feeding z_q into the decoder, CA achieves better results than Concat. Removing either $\mathrm{Dec}_{\mathrm{G}}$ or Dec_{ϵ} results in inferior performance compared to their combined usage, with $\mathrm{Dec}_{\mathrm{G}}$ contributing more. Moreover, though the models w/o MED perform well in all piece-level metrics, their bar-level metrics are generally lower than those of MusER.

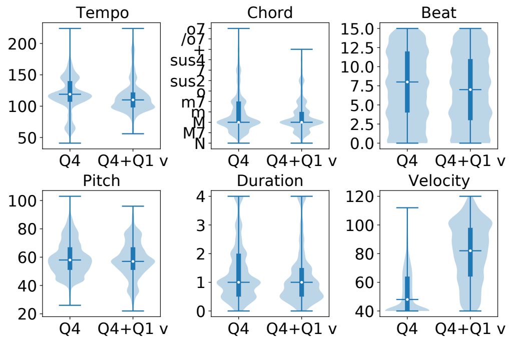


Figure 4: Evaluations on musical element transfer. (Q4+Q1 v) means the latent vectors of velocity in Q1 are concatenated with the latent vectors of other elements in Q4.

Comparison with Previous Models. Table 2 also shows the comparative results between MusER and the previous models. We observe that the music generated by MusER is not significantly different from the real data in three out of the six metrics, surpassing the compared models. However, it demonstrates inferiority to Transformer GAN concerning PR and exhibits lower proficiency in learning B-POLY.

Musical Element Transfer Performance

For evaluating the performance of element transfer, we transfer velocity, which plays a prominent role in distin-

Model	Humanness	p-value	Richness	p-value	Overall	p-value
EMOPIA (Real data)	4.03±1.01	-	3.80±1.08	-	3.95±0.97	-
CP Transformer	3.25±0.94	1.8e-06	3.35±0.88	**0.0557**	3.38±0.94	0.0005
Transformer GAN	3.05±0.97	1.1e-08	2.88±0.81	4.6e-06	3.08±0.82	6.8e-08
MusER (ours)						
Non-transferred music	**4.00±0.87**	**0.1116**	**3.55±0.89**	**0.7914**	**3.78±0.79**	**0.0893**
Transferred music	3.20±0.93	1.6e-07	3.43±0.77	**0.1257**	3.30±0.81	2.4e-05

Table 3: Results of the listening survey. Mann-Whitney U test is adopted to determine whether there is a statistical difference.

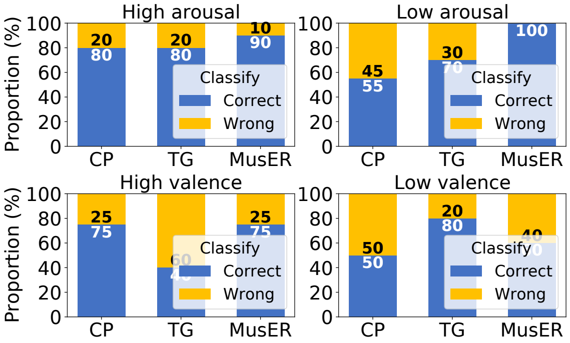


Figure 5: The emotion classification results when the target emotion is high arousal, low arousal, high valence, and low valence. The acronyms CP and TG correspond to the CP Transformer and Transformer GAN.

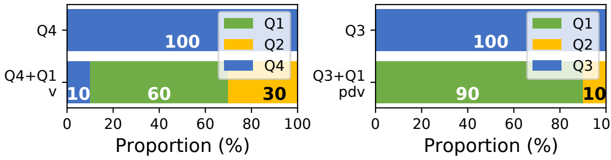


Figure 6: The emotion classification results for music clips after element transfer.

guishing arousal. Specifically, we randomly selected 20 music clips each from Q1 and Q4 in the dataset. Next, we concatenated the latent vectors of velocity from Q1 with the latent vectors of other music elements from Q4. The resulting latent vectors were fed into the decoder to generate music. Figure 4 depicts the element distributions of the original music in Q4 and the velocity-transferred music, revealing a significant change in velocity distribution while the distributions of other elements remained relatively unchanged.

Subjective Listening Test

We conduct human listening tests to subjectively evaluate the quality of the generated music and assess whether humans could perceive the given emotions from the generated music. Additionally, we investigate the potential of musical element transfer in facilitating emotional change.

A total of 20 participants containing 10 males and 10 females were asked to rate the generated music on a five-point Likert scale in terms of three criteria (Hung et al. 2021): **Humanness**, **Richness**, **Overall**, and make binary judgments on the emotions conveyed by the generated music samples, namely 1) **Valence** (V): is the music negative or positive; 2) **Arousal** (A): is the music calming or exciting.

Table 3 shows the subjective scores for real music and music generated by various models. It is concluded that music generated by MusER when not performing element transfer (i.e., non-transferred music) gets the best scores. Music generated via element transfer (i.e., transferred music) slightly outperforms music generated by Transformer GAN but falls short of music generated by CP Transformer and the non-transferred music. We conjecture that this is because the model has not learned how to harmoniously organize the latent vectors of elements belonging to distinct emotions.

Figure 5 illustrates the results on emotion controllability of different models. The results reveal that all models achieved a correct rate of over 50% in generating the given A, with MusER achieving the highest accuracy followed by Transformer GAN. As for V, the samples generated by Transformer GAN are easily recognized as having low V, leading to subpar classification results for high V. Our model outperforms the CP Transformer in generating high V.

Last but not least, we contend that transferring the emotion-distinguishable element may lead to a corresponding change in musical emotion. To initially investigate this, participants were asked to classify the emotion of the transferred music. The results are presented in Figure 6. Remarkably, we observe that when the latent vectors of velocity in Q4 were substituted with those in Q1, 90% of participants perceived an alteration in A. Similarly, after exchanging the latent vectors of pitch (p), duration (d), and velocity (v) in Q3 with those in Q1, the arousal transfer was accomplished, with 90% of participants even perceiving a change in V.

Conclusion

In this paper, we propose MusER for disentangling musical elements, investigating their contributions to identifying emotion and generating emotional music. MusER incorporates a musical element disentanglement module (MED) with a regularization loss that disentangles the latent space to accommodate different element sequences and a two-level decoding module (TD) to take full advantage of latent variables with distinct semantics. The experimental results demonstrated that MusER outperformed previous methods for generating emotional music in both objective and subjective evaluations while yielding a disentangled latent space. We believe our contributions will catalyze further advancements in the intelligent manipulation of musical elements.

References

Akama, T. 2019. Controlling Symbolic Music Generation based on Concept Learning from Domain Knowledge. In *Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019, Delft, The Netherlands, November 4-8, 2019*, 816–823.

Bao, C.; and Sun, Q. 2022. Generating Music with Emotions. *IEEE Transactions on Multimedia*.

Chen, K.; Wang, C.; Berg-Kirkpatrick, T.; and Dubnov, S. 2020a. Music SketchNet: Controllable Music Generation via Factorized Representations of Pitch and Rhythm. In *Proceedings of the 21th International Society for Music Information Retrieval Conference, ISMIR 2020, Montreal, Canada, October 11-16, 2020*, 77–84.

Chen, K.; Xia, G.; and Dubnov, S. 2020b. Continuous Melody Generation via Disentangled Short-Term Representations and Structural Conditions. In *IEEE 14th International Conference on Semantic Computing, ICSC 2020, San Diego, CA, USA, February 3-5, 2020*, 128–135. IEEE.

Dong, H.; Chen, K.; McAuley, J. J.; and Berg-Kirkpatrick, T. 2020. MusPy: A Toolkit for Symbolic Music Generation. In *Proceedings of the 21th International Society for Music Information Retrieval Conference, ISMIR 2020, Montreal, Canada, October 11-16, 2020*, 101–108.

Ferreira, L.; Lelis, L.; and Whitehead, J. 2020. Computer-generated music for tabletop role-playing games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 16, 59–65.

Ferreira, L. N.; Mou, L.; Whitehead, J.; and Lelis, L. H. 2022. Controlling Perceived Emotion in Symbolic Music Generation with Monte Carlo Tree Search. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 18, 163–170.

Grekow, J. 2021. Generating Musical Sequences with a Given Emotion. In *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 1941–1946. IEEE.

Hadjeres, G.; Nielsen, F.; and Pachet, F. 2017. GLSR-VAE: Geodesic latent space regularization for variational autoencoder architectures. In *2017 IEEE symposium series on computational intelligence (SSCI)*, 1–7. IEEE.

Hawthorne, C.; Elsen, E.; Song, J.; Roberts, A.; Simon, I.; Raffel, C.; Engel, J. H.; Oore, S.; and Eck, D. 2018. Onsets and Frames: Dual-Objective Piano Transcription. In *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*, 50–57.

Holtzman, A.; Buys, J.; Du, L.; Forbes, M.; and Choi, Y. 2020. The Curious Case of Neural Text Degeneration. In *8th International Conference on Learning Representations, ICLR*.

Hsiao, W.-Y.; Liu, J.-Y.; Yeh, Y.-C.; and Yang, Y.-H. 2021. Compound word transformer: Learning to compose full-song music over dynamic directed hypergraphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 178–186.

Hung, H.-T.; Ching, J.; Doh, S.; Kim, N.; Nam, J.; and Yang, Y.-H. 2021. EMOPIA: A Multi-Modal Pop Piano Dataset For Emotion Recognition and Emotion-based Music Generation. In *Proceedings of the 22nd International Society for Music Information Retrieval Conference, ISMIR 2021, Online, November 7-12, 2021*, 318–325.

Ji, S.; and Yang, X. 2023. EmoMusicTV: Emotion-conditioned Symbolic Music Generation with Hierarchical Transformer VAE. *IEEE Transactions on Multimedia*.

Ji, S.; Yang, X.; and Luo, J. 2023. A Survey on Deep Learning for Symbolic Music Generation: Representations, Algorithms, Evaluations, and Challenges. *ACM Comput. Surv.*

Katharopoulos, A.; Vyas, A.; Pappas, N.; and Fleuret, F. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, 5156–5165. PMLR.

Luo, J.; Yang, X.; Ji, S.; and Li, J. 2020. MG-VAE: Deep Chinese folk songs generation with specific regional styles. In *Proceedings of the 7th Conference on Sound and Music Technology (CSMT) Revised Selected Papers*, 93–106. Springer.

Makris, D.; Agres, K. R.; and Herremans, D. 2021. Generating lead sheets with affect: A novel conditional seq2seq framework. In *2021 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.

Neves, P.; Fornari, J.; and Florindo, J. 2022. Generating music with sentiment using Transformer-GANs. In *Proceedings of the 23rd International Society for Music Information Retrieval Conference, ISMIR*, 717–725.

Pati, A.; and Lerch, A. 2019. Latent space regularization for explicit control of musical attributes. In *ICML Machine Learning for Music Discovery Workshop (ML4MD), Extended Abstract, Long Beach, CA, USA*.

Pati, A.; and Lerch, A. 2021. Attribute-based regularization of latent spaces for variational auto-encoders. *Neural Computing and Applications*, 33: 4429–4444.

Pratt, R. R.; Abel, H.-H.; and Skidmore, J. 1995. The effects of neurofeedback training with background music on EEG patterns of ADD and ADHD children. *International Journal of Arts Medicine*.

Ren, Y.; He, J.; Tan, X.; Qin, T.; Zhao, Z.; and Liu, T.-Y. 2020. Popmag: Pop music accompaniment generation. In *Proceedings of the 28th ACM international conference on multimedia*, 1198–1206.

Rousseeuw, P. J. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20: 53–65.

Russell, J. A. 1980. A circumplex model of affect. *Journal of personality and social psychology*, 39(6): 1161.

Van Den Oord, A.; Vinyals, O.; et al. 2017. Neural discrete representation learning. *Advances in neural information processing systems*, 30.

Wang, Q.; Zhang, S.; and Zhou, L. 2023. Emotion-Guided Music Accompaniment Generation Based on Variational Autoencoder. *arXiv preprint arXiv:2307.04015*.

Wang, Z.; Wang, D.; Zhang, Y.; and Xia, G. 2020. Learning Interpretable Representation for Controllable Polyphonic Music Generation. In *Proceedings of the 21th International Society for Music Information Retrieval Conference, ISMIR 2020, Montreal, Canada, October 11-16, 2020*, 662–669.

Wei, S.; and Xia, G. 2021. Learning long-term music representations via hierarchical contextual constraints. In *Proceedings of the 22nd International Society for Music Information Retrieval Conference, ISMIR 2021, Online, November 7-12, 2021*, 738–745.

Yang, R.; Wang, D.; Wang, Z.; Chen, T.; Jiang, J.; and Xia, G. 2019. Deep Music Analogy Via Latent Representation Disentanglement. In *Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019, Delft, The Netherlands, November 4-8, 2019*, 596–603.