

RESEARCH ARTICLE

Semi-supervised emotion-driven music generation model based on category-dispersed Gaussian Mixture Variational Autoencoders

Zihao Ning¹, Xiao Han², Jie Pan^{1*}

1 Communication University of China, Nanjing, China, **2** Nanjing University of Aeronautics and Astronautics, Nanjing, China

* panjie@cucn.edu.cn



OPEN ACCESS

Citation: Ning Z, Han X, Pan J (2024) Semi-supervised emotion-driven music generation model based on category-dispersed Gaussian Mixture Variational Autoencoders. PLoS ONE 19(12): e0311541. <https://doi.org/10.1371/journal.pone.0311541>

Editor: Xiaohui Zhang, Bayer Crop Science United States: Bayer CropScience LP, UNITED STATES OF AMERICA

Received: August 21, 2024

Accepted: September 19, 2024

Published: December 30, 2024

Copyright: © 2024 Ning et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: The data underlying the results presented in the study are available from https://opendatalab.com/OpenDataLab/Lakh_Pianoroll_Dataset.

Funding: The author(s) received no specific funding for this work.

Competing interests: The authors have declared that no competing interests exist.

Abstract

Existing emotion-driven music generation models heavily rely on labeled data and lack interpretability and controllability of emotions. To address these limitations, a semi-supervised emotion-driven music generation model based on category-dispersed Gaussian mixture variational autoencoders is proposed. Initially, a controllable music generation model is introduced, which disentangles and manipulates rhythm and tonal features, enabling controlled music generation. Building on this, a semi-supervised model is developed, leveraging a category-dispersed Gaussian mixture variational autoencoder to infer emotions from the latent representations of rhythm and tonal features. Finally, the objective loss function is optimized to enhance the separation of distinct emotional clusters. Experimental results on real-world datasets demonstrate that the proposed method effectively separates music with different emotions in the latent space, thereby strengthening the association between music and emotions. Additionally, the model successfully disentangles and separates various musical features, facilitating more accurate emotion-driven music generation and emotion transitions through feature manipulation.

1. Introduction

Music, as a significant form of artistic expression, reflects a series of human-specific cognitive patterns. Similar to natural language, music conveys content and emotions through sequential information. Unlike natural language, however, emotion is a crucial component of the information conveyed by music [1]. The emotions evoked by music are not only closely related to its inherent structure but are also induced by the sequential propagation of musical elements over time. Therefore, the automatic generation of music must not only adhere to the rules of music theory but also incorporate emotional elements to inspire creativity [2]. Emotion-driven music generation has wide applications, including the creation of background music for specific scenes in films or games, and in music-based psychotherapy [3].

In recent years, significant advancements have been made in automatic music generation technology, with numerous generative models capable of effectively learning the structural features of music to produce rich musical samples [4]. However, fewer studies have focused on incorporating emotional elements as a condition for music generation. A typical approach to emotion-driven music generation involves using sequence-to-sequence (Seq2Seq) models, such as Recurrent Neural Networks (RNNs) and Transformers, to simultaneously encode both music sequences and emotion labels [5]. This allows the models to learn the feature representations of specific emotions, which are then decoded to generate music corresponding to the targeted emotions. Alternatively, emotion classifiers may be used post-generation to guide the emotional content of the music. Zhao et al. [6] constructed an emotion-driven music generation model using a biaxial LSTM architecture [7] combined with a Lookback module. Ferreira et al. [8] encoded music as a series of textual events and utilized multiplicative LSTM networks to create music with specific emotional attributes. To further enhance the model's ability to capture long-term dependencies in the structural context of music, pre-trained Transformer models were employed in the works of Hung et al. [9] and Sulun et al. [10] to achieve emotion-driven music generation. In these models, emotional information, like the input musical events, is embedded as tokens within the model. To obtain better semantic representations of emotions, some studies utilized Variational Autoencoders (VAEs) [11] to learn emotion label embeddings and model the latent space of the data [12]. This approach maps the data distribution into a low-dimensional continuous space, from which diverse and continuous samples can be generated. Grekow et al. [13] proposed using Conditional Variational Autoencoders (CVAE) to generate music with specific emotions, leveraging Gated Recurrent Units (GRU) [14] to construct the encoder and decoder. Huang et al. [15] further combined CVAE with GANs to generate more accurate emotional music. In this model, a discriminator distinguishes the authenticity of the music generated by the CVAE decoder, while an additional emotion classifier determines the emotional category of the output music, thus aligning the generated music more closely with the target emotion. However, the aforementioned methods rely on fully supervised training and generate emotion-specific music directly based on labels, which presents two significant challenges. First, fully supervised training excessively depends on labeled data, yet the music domain lacks abundant standardized emotional datasets. Moreover, emotion is highly subjective, leading to considerable label noise during data annotation. Second, emotion labels alone are insufficient to fully capture the emotional characteristics of music. Generating music directly from labels overlooks the intricate relationships between emotions and musical elements, rendering the generated music less interpretable in terms of its underlying emotional representation.

Recent research, such as Music FaderNets [16], utilized Gaussian Mixture Variational Autoencoders (GMVAE) [17] in a semi-supervised approach to learn latent space representations of note and rhythm features along the Arousal dimension of Russell's two-dimensional emotion space [18]. These features were then employed to control the Arousal dimension, enabling the generation of emotion-specific music. GMVAE extends the unimodal Gaussian distribution in VAE to a mixture of Gaussians with multiple components, allowing the latent space to be partitioned into distinct clusters. This enables direct inference of data categories from latent variables without the need for additional neural networks to learn category information, demonstrating high performance in semi-supervised generative tasks. Luo et al. [19] proposed a GMVAE-based audio music generation model that disentangled and learned latent spaces for different categories of pitch and timbre. When generating music with specific pitch or timbre, the corresponding latent variables were resampled from the target category's latent space and decoded into the output. Tan et al. [20] also employed GMVAE in the Music FaderNets model to achieve controllable generation of MIDI music notes and rhythms. However,

GMVAE has certain limitations, including the potential for mode collapse [21]. When data is overly similar, the Gaussian components may not separate well during training, making it difficult to distinguish between different categories, thus impacting the quality of the learned representations and generated outputs. Furthermore, most existing VAE-based music generation models rely on Recurrent Neural Networks (RNNs) as encoders and decoders [22]. Given that music is a long-sequence data type, RNNs are limited in their ability to model such sequences, often leading to the loss of contextual dependencies and issues like gradient vanishing or explosion.

To address these challenges, this paper proposes a semi-supervised emotion-driven music generation model based on Category-Dispersed Gaussian Mixture Variational Autoencoders (CDGMVAE). This model is designed to learn the influence of different musical features on emotions, generate music with long-term dependency structures and specific emotions, and achieve emotion transitions by manipulating musical features. The specific contributions of this paper are as follows:

- A semi-supervised emotion-driven music generation model based on GVAE with variance penalties and mutual information enhancement is proposed. This model ensures better separation of different emotional music in the latent space, strengthens the correlation between music and emotional information, aligns the generated music more closely with the target emotion, and improves the robustness and generalization ability of the semi-supervised model.
- A feature disentanglement mechanism constrained by independent encoder constraints and generative adversarial loss functions is proposed to separate and learn the latent variable representations of rhythm and tonal features from music sequences. By manipulating these two feature representations, the generation and transformation of specific emotional music can be achieved.
- The Transformer-XL network [23] is introduced as the encoder and decoder for GMVAE. Its segment-level recurrence mechanism and relative positional encoding effectively capture the longer contextual dependencies in music sequences and enhance the model's ability to focus on different features, thereby improving the model's expressiveness.

2. Disentangled controllable generation with Variational Autoencoders

Variational Autoencoders (VAEs) combine latent variable models with deep generative models, offering powerful representation and generation capabilities. By incorporating a disentanglement representation mechanism, VAEs can learn representations of multiple features, imbuing latent variables with more meaningful semantic interpretations. Due to the manipulability of the latent space, VAEs are effective models for achieving controllable music generation. Building on the principles of the VAE model, this paper proposes a controllable music generation model that manipulates latent variable representations to alter the structure of music, referred to as Control-VAE. The model structure is illustrated in [Fig 2](#), comprising an encoder, a latent space, and a decoder. Given an input sequence $\mathbf{x} = [\mathbf{Bar}_1, \mathbf{Bar}_2, \dots, \mathbf{Bar}_T]$ containing T consecutive bars of music, the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ learns the latent variable representations \mathbf{z} of each bar through posterior inference. These representations are then used by the decoder $p_\theta(\mathbf{x}|\mathbf{z})$ to reconstruct the input data, producing a new sample $\tilde{\mathbf{x}}$. The model can be optimized by maximizing the evidence lower bound (ELBO) of the log-likelihood function $p(\mathbf{x})$, as shown in Eq (1):

$$\log p(\mathbf{x}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{KL}[q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})] = \mathcal{L}_{ELBO} \quad (1)$$

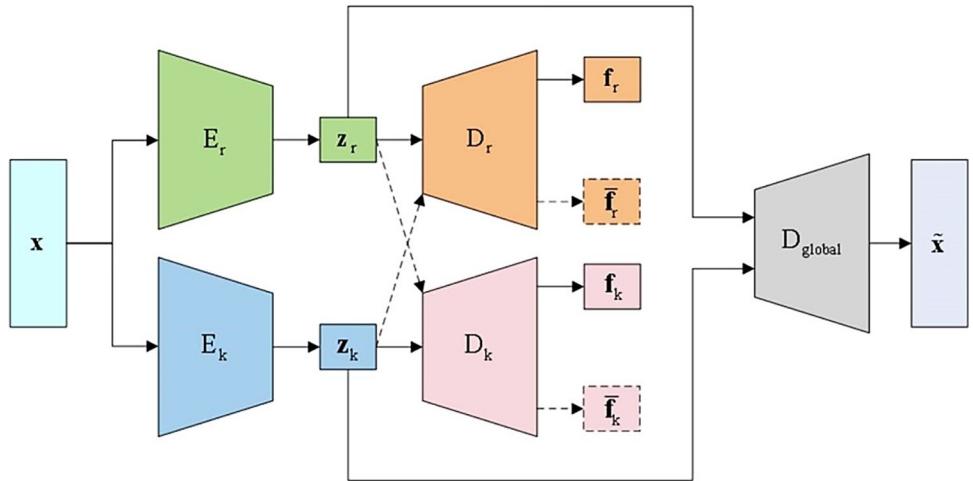


Fig 1. Structure of music feature disentanglement.

<https://doi.org/10.1371/journal.pone.0311541.g001>

where $\mathbb{E}[\cdot]$ denotes the expectation, representing the need to maximize the probability of generating real data, which can be achieved by minimizing the reconstruction loss between the input $|x$ and the output \tilde{x} . The term $D_{KL}[\cdot \parallel \cdot]$ represents the KL divergence that needs to be minimized between the latent distribution $q_\phi(\mathbf{z}|\mathbf{x})$ and the prior Gaussian distribution $p(\mathbf{z})$.

To achieve disentangled and controllable generation of rhythm and tonal features in music, this paper proposes a feature disentanglement mechanism constrained by independent encoders and adversarial loss functions, as depicted in **Fig 1**. Considering the rhythm features \mathbf{f}_r and tonal features \mathbf{f}_k of music, two identical encoder networks E_r and E_k were trained separately to learn the latent variable representations \mathbf{z}_r and \mathbf{z}_k of the rhythm features \mathbf{f}_r and tonal features \mathbf{f}_k from the original music sequence. Two separate local decoders D_r and D_k were then used to predict the rhythm features \mathbf{f}_r and tonal features \mathbf{f}_k , with the training objective being to minimize the error between the real features and the predicted features. Additionally, adversarial training is incorporated into the model based on the concept of GANs. By feeding inputs \mathbf{z}_r and \mathbf{z}_k as the input to D_r and D_k , incorrect information is fed back to each decoder to produce false predictions $\bar{\mathbf{f}}_r$ and $\bar{\mathbf{f}}_k$, thereby enabling D_r and D_k to adversarially counteract each other during training. This ensured that only the correct feature predictions were made based on the corresponding latent variable inputs, effectively removing irrelevant feature information from each latent variable. Finally, the two latent variables were merged (i.e., $\mathbf{z} = \text{Concat}[\mathbf{z}_r, \mathbf{z}_k]$) and input into the global decoder D_{global} to generate the complete music sequence, which is optimized using the ELBO. According to the proposed method, the objective loss function of the Control-VAE model can be defined as:

$$\mathcal{L}_{\text{ELBO}} = \mathbb{E}_{q_{\phi_r}(\mathbf{z}_r|\mathbf{x})q_{\phi_k}(\mathbf{z}_k|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z}_r, \mathbf{z}_k)] - \mathcal{L}_{KL}^i + \mathcal{L}_{\text{Dis}} \quad (2)$$

where $\mathcal{L}_{KL}^i = D_{KL}[q_{\phi_i}(\mathbf{z}_i|\mathbf{x}) \parallel p(\mathbf{z}_i)]$, i represent the rhythm or tonal features, and \mathcal{L}_{Dis} represents the disentanglement loss, defined as follows:

$$\begin{aligned} \mathcal{L}_{\text{Dis}} = & \mathbb{E}_{q_{\phi_r}(\mathbf{z}_r|\mathbf{x})q_{\phi_k}(\mathbf{z}_k|\mathbf{x})}[\log p_{\phi_r}(\mathbf{f}_r|\mathbf{z}_r)p_{\phi_k}(\mathbf{f}_k|\mathbf{z}_k)] \\ & + \mathbb{E}_{q_{\phi_r}(\mathbf{z}_r|\mathbf{x})q_{\phi_k}(\mathbf{z}_k|\mathbf{x})}[\log(1 - p_{\phi_r}(\mathbf{f}_r|\mathbf{z}_r))(1 - p_{\phi_k}(\mathbf{f}_k|\mathbf{z}_k))] \end{aligned} \quad (3)$$

where the first term represents the loss when correctly predicting the features, and the second term represents the adversarial loss.

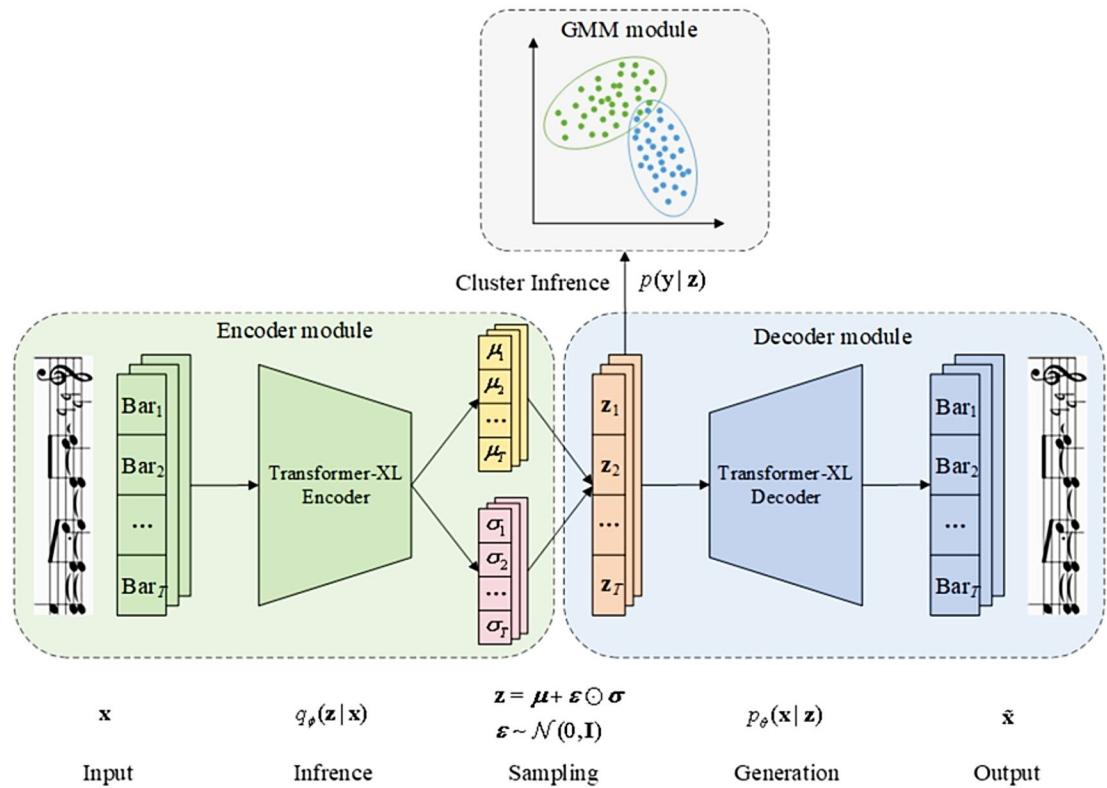


Fig 2. Schematic of the CDGMVAE model structure.

<https://doi.org/10.1371/journal.pone.0311541.g002>

3. Category-dispersed Gaussian mixture variational autoencoder

The structure of our method CDGMVAE is illustrated in Fig 2 and comprises three main modules: (1) The encoder module, which receives the music input sequence segmented by bars and uses a Transformer-XL encoder to learn the mean and variance of the latent distribution; (2) The decoder module, which reparametrizes the latent variable representations sampled from the latent distribution and uses a Transformer-XL decoder to reconstruct the music sequence; and (3) The Gaussian mixture module, which employs a Gaussian mixture model (GMM) [78] to learn the categorical information of latent variables via semi-supervised learning, thereby clustering the music data by emotion.

3.1 Semi-supervised Gaussian mixture variational autoencoder

GMVAE integrates a VAE with a GMM, enabling the extension of the continuous latent space of a VAE into a discrete space with multiple Gaussian components. Each Gaussian component represents a cluster that contains latent variable representations of the same category. The CDGMVAE model leverages this property to infer the emotional category of the latent variable representations of music features using a semi-supervised approach, projecting them onto the corresponding Gaussian components. During the generation phase, latent variables are sampled from the Gaussian component of the target category, and the corresponding emotional music is generated by decoding these latent variables. The semi-supervised GMVAE can guide the model to learn the feature representations and boundaries of each category using a small amount of labeled data as supervision signals, while the unlabeled data provides a broader data

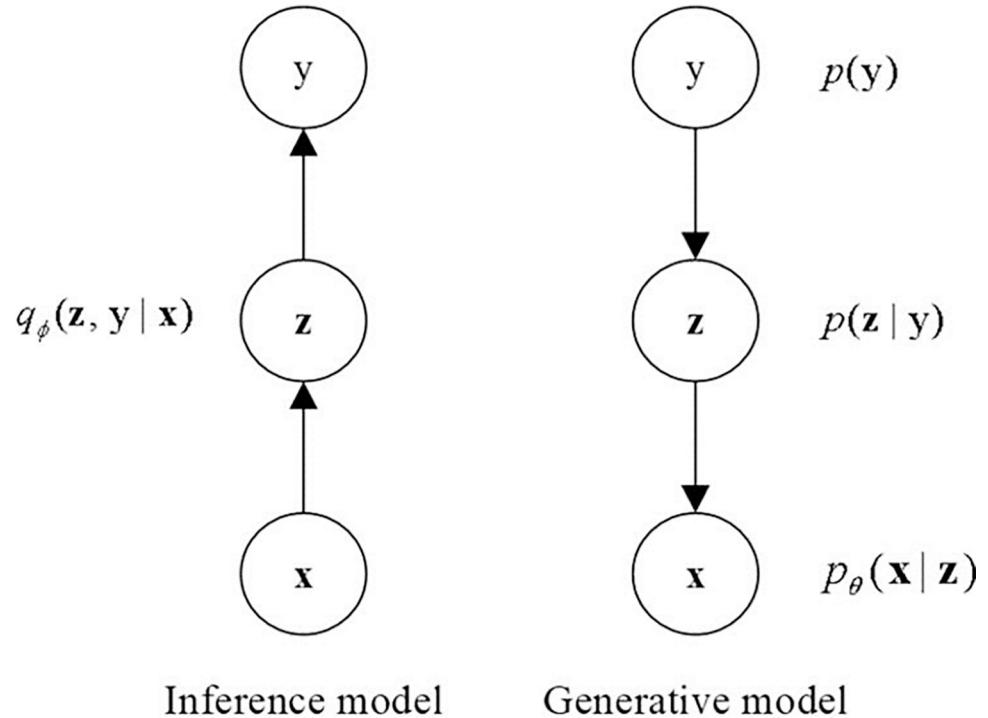


Fig 3. Probabilistic graphical model of GMVAE.

<https://doi.org/10.1371/journal.pone.0311541.g003>

distribution. By leveraging the labeled data, similar samples can be grouped into the same category without the need to introduce additional neural network classifiers.

In implementation, the GMVAE can be divided into an inference network and a generative network, with its probabilistic graphical model illustrated in Fig 3. For the inference network, given the music input sequence x , the encoder learns the latent variable representations z of any feature and introduces a classification variable y to infer the emotional category z . For the generative network, latent variables z are sampled from the Gaussian component of the target category based on the classification variable y , and the decoder reconstructs the input data x based on these latent variables. The generative process can be defined by the joint probability distribution as shown in Eq (4):

$$p_\theta(x, z, y) = p_\theta(x|z)p(z|y)p(y) \quad (4)$$

where $p(y) = \text{Cat}(y|1/K)$ represents the categorical distribution with K categories. $p(z|y) = N(z|\mu_y, \sigma_y^2)$ denotes the latent distribution of a specific Gaussian component within a category, characterized by a learnable mean μ_y and variance σ_y^2 . $q_\phi(x|z)$ is a neural network parameterized by θ , which is responsible for decoding and generating sample data. For the inference process, GMVAE approximates the true posterior distribution $p(z,y|x)$ through variational inference $q_\phi(y|x)$, similar to VAE. According to the mean-field approximation theory [24], $q_\phi(y|x)$ can be further factorized as shown in Eq (5):

$$q_\phi(z, y|x) = q_\phi(z|x)q_\phi(y|x) \quad (5)$$

where $q_\phi(z|x) = N(z|\mu_x, \sigma_x^2)$ represents the latent distribution learned by the encoder network parameterized by ϕ , which gradually approximates the Gaussian distribution $p(z|y)$ under the corresponding category during training. Similarly, $q_\phi(y|x)$ is used to learn the categorical

information from the input x , but this requires an additional neural network for fitting, thereby introducing more parameters to the model and increasing dependency on labeled data. To mitigate this, $p(y|z)$ can be approximated as $q_\phi(y|x)$ following the derivation process in VaDE [25], as shown in Eq (6):

$$q_\phi(y|x) = \mathbb{E}_{q_\phi(z|x)}[p(y|z)] \approx \frac{1}{N} \sum_{n=1}^N \frac{p(z|y)p(y)}{\sum_{\hat{y}=1}^K p(z|\hat{y})p(\hat{y})} \quad (6)$$

This method leverages the properties of GMM and uses Bayesian inference to determine which Gaussian component's latent distribution generates the data, thus identifying its category without introducing additional neural networks. Finally, Eq (6) is approximated through Monte Carlo estimation of the expectation $q_\phi(y|x)$, where N represents the number of samples used for Monte Carlo estimation. In practice, the mean μ_y and variance σ_y^2 of each Gaussian component can be initialized using data with known categories. These parameters are then refined during the training process until they accurately reflect the categorical information. Unlabeled data are also utilized to update the parameters of the Gaussian components, further enhancing the model's ability to learn categories.

Similar to VAE, GMVAE optimizes model parameters by maximizing the ELBO. Based on the aforementioned generative and inference processes, the log-likelihood function $\log p(x)$ can be defined as shown in Eq (7):

$$\log p(x) \geq \mathbb{E}_{q_\phi(z|x)}[\log \frac{p_\theta(x, z, y)}{q_\phi(z, y|x)}] = \mathcal{L}_{ELBO} \quad (7)$$

Under the semi-supervised approach, which uses both labeled and unlabeled data, \mathcal{L}_{ELBO} can be further expanded as shown in Eq (8):

$$\begin{aligned} \mathcal{L}_{ELBO} &= \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \begin{cases} \mathcal{L}_{KL}^{unsup}, & \text{unsupervised} \\ \mathcal{L}_{KL}^{sup}, & \text{supervised} \end{cases} \\ \mathcal{L}_{KL}^{unsup} &= \mathbb{E}_{q_\phi(y|x)}[\text{KL}[q_\phi(z|x) \| p(z|y)]] + \text{KL}[q_\phi(y|x) \| p(y)] \\ \mathcal{L}_{KL}^{sup} &= \text{KL}[q_\phi(z|x) \| p(z|y)] \end{aligned} \quad (8)$$

where $\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$ represents the probability of generating real data, which can be achieved by minimizing the cross-entropy reconstruction loss. \mathcal{L}_{KL}^{unsup} represents the KL divergence loss under unsupervised learning, which requires minimizing the KL divergence between the latent distribution learned by the encoder and the latent distribution under the true category, as well as between the inferred posterior distribution of the category and the assumed uniform categorical distribution. \mathcal{L}_{KL}^{sup} represents the KL divergence loss under supervised learning, where only the former KL divergence needs to be minimized.

3.2 Variance penalization and mutual information enhancement

When applying the GMVAE model to perform clustering inference on data, high data similarity and an insufficient number of Gaussian mixture components can hinder the learning of more complex distributions. In the latent space, each Gaussian component may struggle to separate during training, often resulting in components with closely aligned means and variances, and in extreme cases, multiple Gaussians may collapse into a single Gaussian. This phenomenon, known as mode collapse, is a significant challenge for GMVAE in distinguishing between different categories of data.

To alleviate the mode collapse issue in GMVAE, we further analyze the ELBO from Section 3.1 and identify the KL divergence regularization term as a primary contributor to this problem. Therefore, the KL divergence loss term $\text{KL}_z = \mathbb{E}_{q_\phi(y|x)}[\text{KL}[q_\phi(z|x) \| p(z|y)]]$ in the latent distribution $\mathcal{L}_{\text{KL}}^{\text{unsup}}$ is re-expressed based on Eq (8) as:

$$\begin{aligned} \text{KL}_z &= \sum_{\hat{y}=1}^K q_\phi(\hat{y}|x) \mathbb{E}_{q_\phi(z|x)} \log \frac{q_\phi(\hat{y}|x)}{p(z|\hat{y})} \\ &= \mathbb{E}_{q_\phi(z|x)} \log \frac{q_\phi(\hat{y}|x)}{\prod_{\hat{y}=1}^K p(z|\hat{y})^{q_\phi(\hat{y}|x)}} \end{aligned} \quad (9)$$

where $\mathbb{E}_{q_\phi(y|x)} = \sum_{\hat{y}=1}^K q_\phi(\hat{y}|x) = 1$ denotes the Gaussian component under category \hat{y} , characterized by mean μ_y and variance σ_y^2 . Assuming the variances of all Gaussian components are equal, the Gaussian distribution's probability density function $p(z|\hat{y})$ can be expressed as:

$$p(z|\hat{y}) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(z - \mu_y)^2}{2\sigma_y^2}\right) \quad (10)$$

Let $\prod_{\hat{y}=1}^K p(z|\hat{y})^{q_\phi(\hat{y}|x)}$ represent $f(x, \hat{y}, z)$, and combining with Eq (10), it can be represented as:

$$\begin{aligned} f(x, \hat{y}, z) &= \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{1}{2\sigma_y^2} \sum_{\hat{y}=1}^K q_\phi(\hat{y}|x) (z - \mu_y)^2\right) \\ &= \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left[-\frac{1}{2\sigma_y^2} (z - \mathbb{E}_{q_\phi(y|x)} \mu_y)^2\right] \\ &\quad \cdot \exp\left[-\frac{1}{2\sigma_y^2} (\mathbb{E}_{q_\phi(y|x)} \mu_y^2 - \mathbb{E}_{q_\phi(y|x)}^2 \mu_y)\right] \end{aligned} \quad (11)$$

The first factor of the product can be regarded as the average Gaussian component across all categories $p(z|y)$, while the second factor represents the variance of the means of all Gaussian components, which measures the dispersion of these components. Combining Eqs (9) and (11), KL_z can be transformed, as shown in Eq (12):

$$\text{KL}_z = \text{KL}[q_\phi(z|x) \| p(z|y)] + \frac{1}{2\sigma_y^2} (\mathbb{E}_{q_\phi(y|x)} \mu_y^2 - \mathbb{E}_{q_\phi(y|x)}^2 \mu_y) \quad (12)$$

When maximizing GMVAE's ELBO during training, the variance term factor in KL_z also gets minimized, causing the variance of the means of Gaussian components to shrink, leading to tighter connections between Gaussian components in the latent space and impairing the model's ability to correctly distinguish between categories, thereby causing mode collapse. To reduce this contraction trend, we introduce a penalization hyperparameter α , $\alpha \in [0, 1]$ before the variance term in KL_z :

In addition to improving the separability of Gaussian components to enhance clustering accuracy, establishing a strong connection between the input data and category information is essential. However, research has shown that VAE's ELBO has an information suppression term that weakens the relationship between input data and latent variables [26]. This bottleneck also exists between the input data and category information in GMVAE. During model

training, the average ELBO of a batch of data is typically computed for gradient updates. Therefore, combined with Eq (12), \mathcal{L}_{ELBO} can be expanded as shown in Eq (13):

$$\mathcal{L}_{ELBO} = \mathbb{E}_x[\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]] - \text{KL}_z - \text{KL}[q_\phi(y|x)\|p(y)] \quad (13)$$

According to the definition of mutual information concerning KL divergence [27], the mutual information $\text{MI}(y,x)$ between input data x and category information y can be expressed as shown in Eq (14):

$$\text{MI}(y, x) = \mathbb{E}_x[\text{KL}[q_\phi(y|x)\|p(y)]] - \text{KL}[q_\phi(y|x)\|p(y)] \quad (14)$$

Therefore, \mathcal{L}_{ELBO} can be rewritten as shown in Eq (15):

$$\mathcal{L}_{ELBO} = \mathbb{E}_x[\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \text{KL}_z] - \text{MI}(y, x) - \text{KL}[q_\phi(y)\|p(y)] \quad (15)$$

This shows that during training, the mutual information between input data and category information is minimized, reducing the dependence between these two random variables and diminishing clustering accuracy. Therefore, we propose a joint optimization of ELBO and mutual information, defined as:

$$\mathcal{L}_{ELBO} + \text{MI}(y, x) = \mathbb{E}_x[\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \text{KL}_z] - \text{KL}[q_\phi(y|x)\|p(y)] \quad (16)$$

This method still allows for optimizing the learning of the categorical distribution through $\text{KL}[q_\phi(y|x)\|p(y)]$, where $q_\phi(y) = \mathbb{E}_x[q_\phi(y|x)]$ represents the discrete marginal probability distribution of $q_\phi(y|x)$, which can be averaged over data points within a batch, reducing the difference between the categorical distribution across all data points and the prior distribution.

Finally, combining variance penalization and mutual information optimization yields the unsupervised KL divergence loss for GMVAE, as shown in Eq (17):

$$\mathcal{L}_{KL}^{\text{unsup}} = \text{KL}[q_\phi(z|x)\|p(z|y)] + \frac{\alpha}{2\sigma_y^2} (\mathbb{E}_{q_\phi(y|x)}\mu_y^2 - \mathbb{E}_{q_\phi(y|x)}^2\mu_y) + \text{KL}[q_\phi(y|x)\|p(y)] \quad (17)$$

These two enhancement methods ensure that data from different categories are better separated in the latent space, while data from the same category are more compactly grouped.

3.3 CDGMVAE objective function

In supervised learning, the loss function should include not only the corresponding KL divergence loss but also a classification cross-entropy loss function to guide the learning of categories. The classification loss is defined as shown in Eq (18):

$$\mathcal{L}_{label} = -\sum_{c=1}^K y_c \log q_\phi(y_c|x) \quad (18)$$

where K is the total number of categories, y_c represents the true label indicating whether the sample belongs to the c th category, and $q_\phi(y_c|x)$ is the predicted probability for the c th category.

Additionally, by integrating the disentanglement loss \mathcal{L}_{Dis} discussed in Section 2, we can derive the final objective loss function for the CDGMVAE model. Given the latent variable z_i of the music feature f and the category variable y_i , the objective loss function can be described

by Eq (19):

$$\mathcal{L}_{\text{CDGMVAE}} = \mathbb{E}_{q_{\phi_r}(z_r|x)q_{\phi_k}(z_k|x)}[\log p_{\theta}(x|z_r, z_k)] + \mathcal{L}_{\text{Dis}} \\ - \sum_i \begin{cases} \mathcal{L}_{\text{KL}}^{\text{unsup},i}, & \text{unsupervised} \\ \mathcal{L}_{\text{KL}}^{\text{sup},i} - \mathcal{L}_{\text{label}}^i, & \text{supervised} \end{cases} \quad (19)$$

3.4 Encoder and decoder

Traditional VAE models and their variants often use RNNs to construct the encoder and decoder. However, due to RNNs' limited modeling capabilities for long sequence data, the Transformer architecture [28] has also been considered in conjunction with VAE models for music generation, as seen in models like Transformer-VAE [29] and MuseMorphose [30]. However, the Transformer's input sequence length is still constrained by memory, requiring self-attention calculations over the entire sequence each time. To address this issue, the CDGMVAE model adopts the Transformer-XL network structure, as shown in [Fig 4](#). Transformer-XL leverages segment-level recurrence and relative positional encoding to learn the contextual structure across more extended musical bars.

In this model, the music sequence is divided into multiple segments of equal length, denoted as $x = [\text{Bar}_1, \text{Bar}_2, \dots, \text{Bar}_T]$, where T represents the number of bars in each segment. Let $h_{\tau-1}^{n-1} \in \mathbb{R}^{L \times d}$ be the hidden state of the $(\tau-1)$ th bar in the $n-1$ layer of self-attention, where L is the sequence length of each bar, and d is the dimension of the hidden state. When calculating the hidden state h_{τ}^n of the τ th bar in the n th layer, the model considers reusing the hidden state from the previous bar. The calculation process is shown in Eq (20):

$$\begin{aligned} \tilde{h}_{\tau}^{n-1} &= \text{Concat}[\text{SG}(h_{\tau-1}^{n-1}) \circ h_{\tau}^{n-1}] \\ Q_{\tau}^n, K_{\tau}^n, V_{\tau}^n &= h_{\tau}^{n-1} W_Q^T, \tilde{h}_{\tau}^{n-1} W_K^T, \tilde{h}_{\tau}^{n-1} W_V^T \\ h_{\tau}^n &= \text{Transformer-layer}(Q_{\tau}^n, K_{\tau}^n, V_{\tau}^n) \end{aligned} \quad (20)$$

where $\text{SG}(\cdot)$ indicates that the gradient of $h_{\tau-1}^{n-1}$ will not be updated when calculating the next bar, $\text{Concat}(\cdot)$ represents the concatenation of the two hidden state matrices along the bar length, and $W_{Q,K,V}$ are the learnable model parameters. Unlike the Transformer, during the attention score calculation, the key-value matrices K_{τ}^n and V_{τ}^n depend on the hidden state $h_{\tau-1}^{n-1}$ of the previous bar in the $n-1$ layer (as shown by the blue lines in [Fig 4](#)) and the hidden state h_{τ}^{n-1} of the current bar in the same layer (as shown by the gray lines in [Fig 4](#)). This method allows the model to cache only the hidden state of the previous bar in memory, establishing a recurrent connection between segments and extending contextual information across all bars.

Transformer models utilize positional encoding to learn the order and positional relationships within an input sequence. However, this positional encoding is absolute, meaning that the encoding information for the same position across different bars is identical, potentially causing the model to misinterpret the sequence. To address this issue, Transformer-XL introduces a relative positional encoding mechanism, incorporating a parameter-free sinusoidal encoding matrix $R \in \mathbb{R}^{L \times d}$. Each row of this matrix represents the relative distance encoding between two positions. When computing hidden states, this relative distance can be dynamically injected into the attention scores, allowing the model to better distinguish between different positions.

When integrating Transformer-XL with the GMVAE model, the goal is to learn the latent variable representations of the data from the hidden states output by the encoder. According to GMVAE's definition of latent distribution $q(z|x)$, the model first learns the mean and

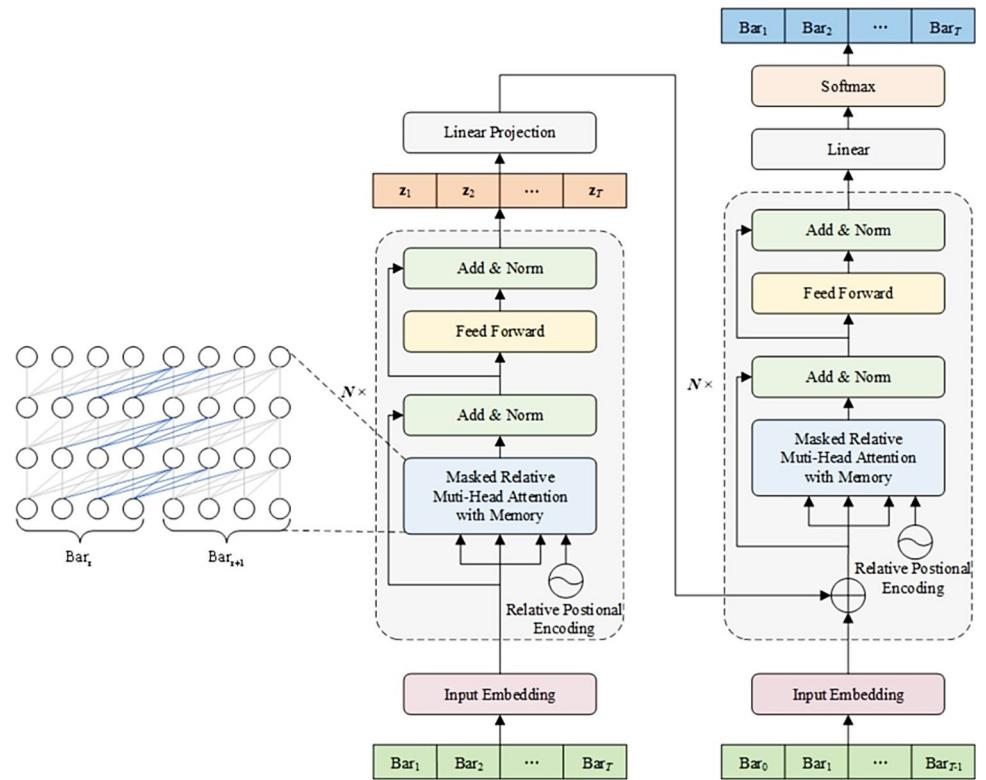


Fig 4. Structure of the CDGMVAE model based on Transformer-XL.

<https://doi.org/10.1371/journal.pone.0311541.g004>

variance of this distribution, then samples the latent variables. This process can be described by Eq (21):

$$\begin{aligned}
 h_{\tau}^{pool} &= \text{Avgpool}([h_{\tau,1}^N, h_{\tau,2}^N, \dots, h_{\tau,L}^N]) \\
 \mu &= h_{\tau}^{pool} W_{\mu}, \log \sigma^2 = h_{\tau}^{pool} W_{\sigma} \\
 z &= \mu + \varepsilon \odot \sigma \varepsilon \sim N(0, I)
 \end{aligned} \tag{21}$$

where $h_{\tau}^{pool} \in \mathbb{R}^d$ represents the overall hidden state of the τ th bar, obtained by average pooling across all positions' hidden states. $\mu, \sigma \in \mathbb{R}^{T \times l}$ denotes the mean and standard deviation vectors of the latent distribution learned from the hidden states, with l representing the set dimension of the latent variable, and $W_{\mu}, W_{\sigma} \in \mathbb{R}^{d \times l}$ is the parameter matrix. $z \in \mathbb{R}^{T \times l}$ is the sampled latent variable from the learned distribution using the reparameterization trick. During the decoding and generation phase, the latent variables are transformed through a linear layer to match the unified model dimension. They are then added to the hidden states output from the previous layer before being input into the subsequent self-attention layer. This method ensures that the latent variable information is effectively utilized to generate the final music sequence.

4. Experimental results and analysis

4.1 MIDI music dataset

In this study, the publicly available Lakh Pianoroll dataset (LPD) [31] is utilized, comprising 174,154 multi-track MIDI music sequences in piano roll format. We selected the LPD-5-full version, excluding sequences with duplicate content, those lacking piano tracks, and those

with a total duration of less than one minute. This process resulted in a dataset of 91,969 sequences, which is split into training, validation, and test sets in an 8:1:1 ratio. Additionally, the dataset is matched with the Million Song Dataset (MSD) [32], enabling the retrieval of emotional metadata for 23,967 music sequences using the metadata provided by MSD and the API services from the Spotify music community (developer.spotify.com). These sequences were annotated in the Russell emotional space with Arousal and Valence values, which were then quantified into two emotional dimensions (high or low) and labeled as A0, A1, V0, and V1, where 0 represents low and 1 represents high. By combining these labels, four corresponding emotional categories were defined: happiness, tension, sadness, and calmness.

4.2 Music data representation

To model MIDI music using the CDGMVAE framework, an appropriate data representation is required. Python libraries such as Pretty_midi and music21 were employed to parse the structural information of MIDI music, and the REMI [33] format is used to encode the music into a sequence of tokenized events. Each event type represents a temporal change in the musical sequence, such as a Bar event indicating the beginning of a measure or a Beat event marking the start of a beat, with corresponding Tempo, Pitch, Velocity, and Duration tokens.

Table 1 outlines the specific meaning of each event type. For the LPD dataset, each music piece can contain up to five tracks. Pitch, Velocity, and Duration are track-specific events, thus denoted as Pitch-[Track], Velocity-[Track], and Duration-[Track] to represent note events on a particular track. An EOS token is appended to the end of each sequence to signify its conclusion, and a PAD token is used for sequence padding, resulting in a total of 923 tokens. During data loading, we randomly selected continuous measures, setting the number of measures T to 20 and the maximum sequence length per measure L to 128.

For disentangled feature learning, rhythm and key features were selected as the musical attributes linked to emotions. The rhythm feature is represented as a $16 \times T$ dimensional vector, defined by the number of simultaneous notes played in each beat. Higher note frequencies within a time span indicate more intense rhythm, reflecting a higher arousal level, whereas lower frequencies correspond to a more moderate and calm rhythm. The key feature is represented as a T -dimensional vector, defined by the possible key signatures within each measure. The keys are categorized into 12 major (e.g., C major) and 12 minor scales (e.g., c minor), with major keys typically conveying positive emotions and minor keys conveying negative emotions, thus corresponding to the valence dimension of emotions. By leveraging these two features within the CDGMVAE model, control over the emotional content of the generated music can be achieved.

Table 1. MIDI music event information.

Event name	Event description	Token count
Bar	Start of a measure	1
Beat	Start of a sub-beat within a measure, with each measure divided into 16 beats	16
Tempo	Tempo of the music at a given time, ranging from 32 to 224 bpm with a step size of 3 bpm	64
Pitch	Pitch of a note, ranging from 21 to 108	88×5
Velocity	Intensity of a note, ranging from 1 to 127 with a step size of 2	64×5
Duration	Duration of a note, quantized as the length in beats, $1 \leq n \leq 16$	16×5
EOS	End of sequence	1
PAD	Padding token	1

<https://doi.org/10.1371/journal.pone.0311541.t001>

4.3 Experimental setup and parameter settings

The CDGMVAE model is developed using the PyTorch deep learning framework and trained on a server equipped with four NVIDIA Tesla P40 GPUs. For the CDGMVAE network structure, the number of layers in both the Transformer-XL encoder and decoder is set to 12. Each self-attention mechanism layer used 8 heads, with hidden layers and latent variable dimensions set to 512 and 128, respectively. To train the Gaussian Mixture Model (GMM) on the Arousal and Valence emotional dimensions, the number of Gaussian components for each dimension is set to 2, representing high and low dimensions. The Xavier initialization method [34] is employed to initialize each Gaussian component's mean vector, with initial standard deviation vectors set to a constant value of e^{-2} . During training, a batch size of 16 is used, and the Adam optimizer is employed for parameter updates, with dynamic learning rate adjustments to facilitate model convergence. For the first 10,000 iterations, a linear warm-up is applied to increase the learning rate from 0 to $1 \times e^{-4}$. Subsequently, in the next 220,000 iterations, a cosine annealing strategy is used to gradually decrease the learning rate to $5 \times e^{-6}$. The model achieved convergence after 50 training epochs.

4.4 Experimental analysis

4.4.1 Semi-supervised generation performance. This section evaluates the semi-supervised generation performance of the CDGMVAE model by comparing it with three fully supervised emotional music generation models and three semi-supervised models. The supervised models include the Transformer-based EMOPIA [35], the GRU-based EmotionBox [36], which links music features with emotions, and the Conditional Variational Autoencoder (CVAE) model [37] that conditions on emotional labels. The semi-supervised models assessed are SSVAE [38], CCVAE [39], and FaderNets [40]. To ensure a fair comparison, all semi-supervised models utilized Transformer-XL for the core network structure, with the exception of SSVAE, which incorporated feature disentanglement mechanisms. Additionally, a music emotion classification model provided by EMOPIA is trained on labeled data from LPD, using 8-bar music sequences as input. After 50 training epochs, the classification performance achieved accuracy comparable to that reported in the original paper. Furthermore, a trained classifier is used to predict the emotional dimensions of Arousal and Valence for all generated music samples. The trained classifier adopts three-layer fully connected layer with parameters of 128-50-10-1. Performance is evaluated using accuracy (Acc), precision (Pre), recall (Recall), and F1 score (F1).

The generation performance of various models under both fully supervised and semi-supervised methods is first compared. Results are shown in Tables 2 and 3, corresponding to the Arousal and Valence emotional dimensions, respectively. Tables 2 and 3 indicate that,

Table 2. Performance comparison of supervised and semi-supervised methods in Arousal dimension.

Models	Supervised				Semi-Supervised			
	Acc/%	Pre/%	Rec/%	F1/%	Acc/%	Pre/%	Rec/%	F1/%
EMOPIA	61.55	58.69	61.52	60.07	-			
EmotionBox	60.06	57.35	58.71	58.02	-			
CVAE	58.63	55.99	56.11	56.05	-			
SSVAE	61.28	58.84	58.73	58.78	73.66	71.94	72.11	72.02
CCVAE	63.74	61.49	61.21	61.35	76.24	75.58	73.07	74.30
FaderNets	68.37	65.21	70.17	67.60	78.65	76.54	78.73	77.62
CDGMVAE	70.52	66.92	73.77	70.18	82.54	79.56	84.60	82.00

<https://doi.org/10.1371/journal.pone.0311541.t002>

Table 3. Performance comparison of supervised and semi-supervised methods in Valence dimension.

Models	Supervised				Semi-Supervised			
	Acc/%	Pre/%	Rec/%	F1/%	Acc/%	Pre/%	Rec/%	F1/%
EMOPIA	57.84	59.81	57.82	58.80	-			
EmotionBox	57.18	59.06	57.66	58.35	-			
CVAE	55.27	59.96	57.34	57.15	-			
SSVAE	58.12	59.96	59.70	59.32	67.15	68.00	69.61	68.80
CCVAE	60.03	61.67	63.43	62.28	69.30	72.88	65.28	68.87
FaderNets	63.45	64.94	64.64	64.79	71.63	72.44	73.38	72.91
CDGMVAE	66.63	68.93	65.28	67.05	75.87	77.37	75.78	76.57

<https://doi.org/10.1371/journal.pone.0311541.t003>

despite all models being trained in a supervised manner, EMOPIA, EmotionBox, and CVAE generally perform worse than other generation models. This can be attributed to the limited contextual learning capabilities of GRU and Transformer compared to Transformer-XL, and the fact that some models use emotional labels directly, lacking representational power compared to feature disentanglement methods. The results also reveal that models trained with semi-supervised methods, incorporating additional unlabeled data, significantly outperform those trained fully supervised across all classification metrics. This demonstrates that VAE-based generation models and their variants effectively utilize unlabeled data to learn broader distribution information, with labeled data providing constraints on categorical information. Overall, CDGMVAE and FaderNets exhibit better generation performance compared to the other two semi-supervised models. This is because the GMVAE model incorporates label information and uses Gaussian mixture distributions to separate the latent space into multiple discrete regions, effectively learning diverse data distributions. In contrast, other semi-supervised VAE models rely on additional neural networks to learn categorical information, which may overly depend on labeled data, limiting the classifier's ability to fully model data distributions. Moreover, the CDGMVAE's enhanced variance and mutual information terms lead to higher classification accuracy compared to FaderNets. This indicates that CDGMVAE better learns latent variable representations for each Gaussian component and accurately distinguishes between different emotional categories, a point further explored in subsequent sections.

To further illustrate the effectiveness of semi-supervised learning, we tested the accuracy of generated music across different amounts of labeled data. Results are presented in Figs 5 and 6, depicting accuracy in the Arousal and Valence emotional dimensions under various supervision rates. Here, $\rho = M/(M+N)$, where M denotes the number of labeled data samples and N represents the total amount of unlabeled data, with $0 < \rho \leq 0.26$. Figs 5 and 6 show that CDGMVAE and FaderNets achieve higher accuracy even with a supervision rate of 0.05. This suggests that GMVAE can learn extensive data distribution across emotional categories with only a small amount of labeled data. Additionally, due to the optimization of variance and mutual information in CDGMVAE, it consistently outperforms FaderNets in accuracy under all supervision conditions. This confirms that the absence of diversity among Gaussian components and insufficient linkage between data and emotional categories can lead to errors in category inference, thereby reducing the emotional quality of generated music. For SSVAE and CCVAE, results indicate that their accuracy is significantly lower than that of GMVAE-based models across all supervision rates, due to their limited ability to learn meaningful latent representations for specific emotions from the classifier. Although SSVAE and CCVAE show a slightly greater increase in accuracy compared to GMVAE models and have considerable potential for improvement, this also highlights their over-reliance on labeled data for

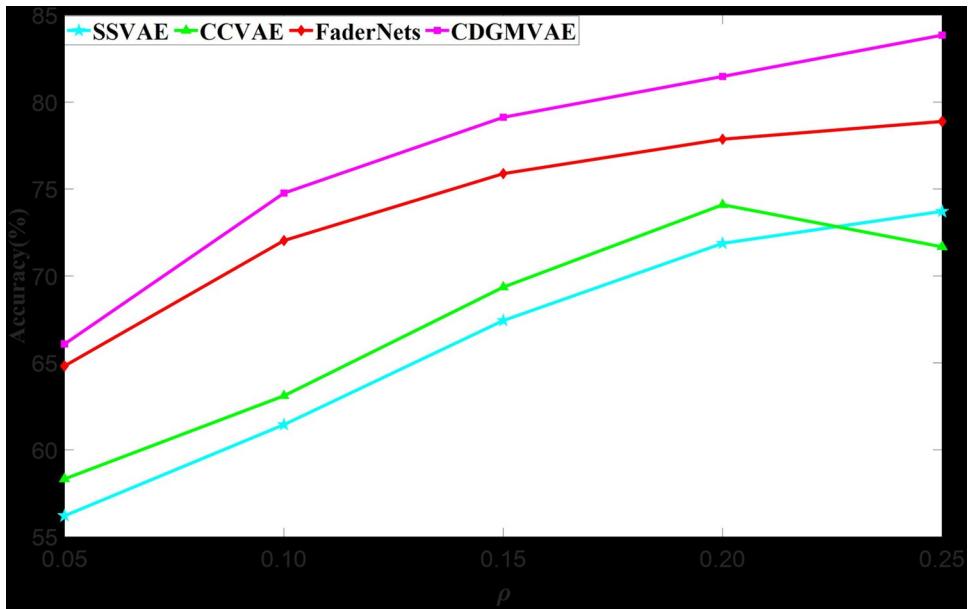


Fig 5. Accuracy of generated music in the Arousal dimension.

<https://doi.org/10.1371/journal.pone.0311541.g005>

supervision signals. Among them, CCVAE exhibits better performance than SSVAE, likely due to its feature disentanglement approach, allowing the classifier to learn emotional categories from rhythm and key features, thereby incorporating the necessary emotional information into latent variables. In contrast, SSVAE directly learns emotional category information from the raw input, which can be less effective in capturing subtle emotional details during the decoding process.

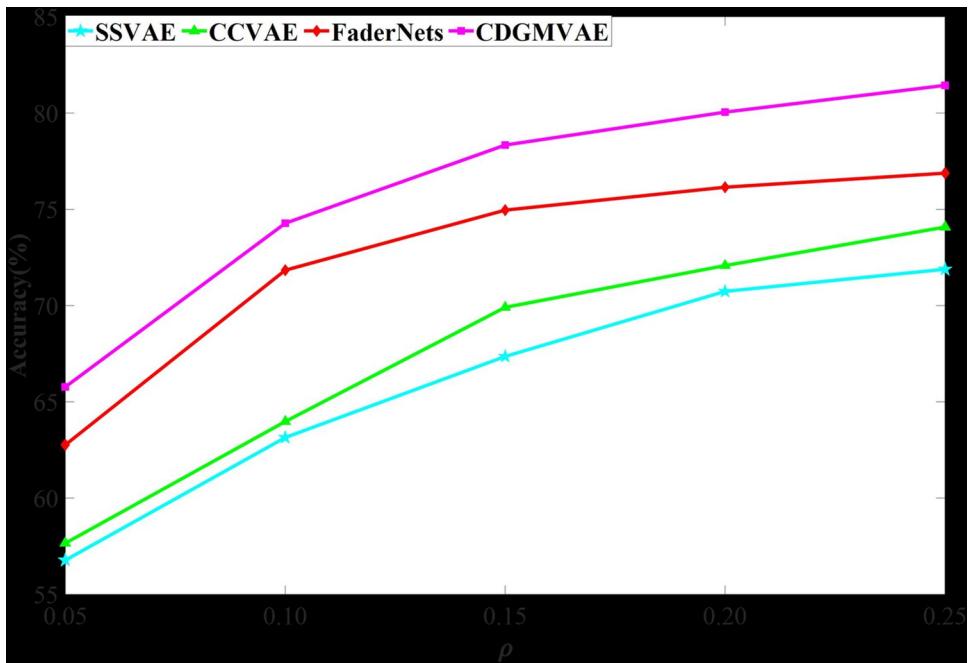


Fig 6. Accuracy of generated music in the valence dimension.

<https://doi.org/10.1371/journal.pone.0311541.g006>

In summary, the music generated by all models is more easily distinguishable in the Arousal dimension compared to the Valence dimension. This is attributed to the more universal and direct nature of rhythm features, which can effectively capture latent representations from raw input data. In contrast, key features require complex music theory inference, which the models may not adequately possess. The experiments also demonstrate that rhythm features are more closely related to the Arousal dimension and are better at influencing the activity level of the music. The Valence dimension involves emotional polarity, which cannot be fully represented by key features alone. This finding aligns with conclusions drawn from the EmotionBox model.

4.4.2 Variance penalty and mutual information enhancement performance analysis.

This subsection discusses how the variance and mutual information penalty terms in the GMVAE evidence lower bound affect the latent space and demonstrates that appropriate correction of these biases enables the model to perform reasonable emotional category inference and generation for music data. Initially, we compared the trends in variance of Gaussian components and their impact on emotional accuracy of generated music under varying penalty weights (i.e., α values). The experimental results are illustrated by the blue lines in Figs 7 and 8. To provide a clearer view of the variations among Gaussian components, we used the T-SNE dimensionality reduction algorithm to project high-dimensional latent variables into a two-dimensional space for visualization. The results are shown in Fig 9.

As analyzed in Section 3.2, lower values of α reduce the influence of the variance term on the overall objective function, resulting in higher variance and classification accuracy. Correspondingly, the distances between Gaussian components in the latent space visualization increase. This enhances the model's ability to generate target emotions in the music effectively. However, when $\alpha \leq 0.2$, while the distances between Gaussian components reach their maximum, the accuracy of emotional classification decreases. The latent space visualization reveals that unnecessary cluster centers form between the two Gaussian components, leading to more frequent misclassifications, with data from one class being assigned to another cluster. This issue arises because excessive penalty strength can cause overfitting, preventing the model

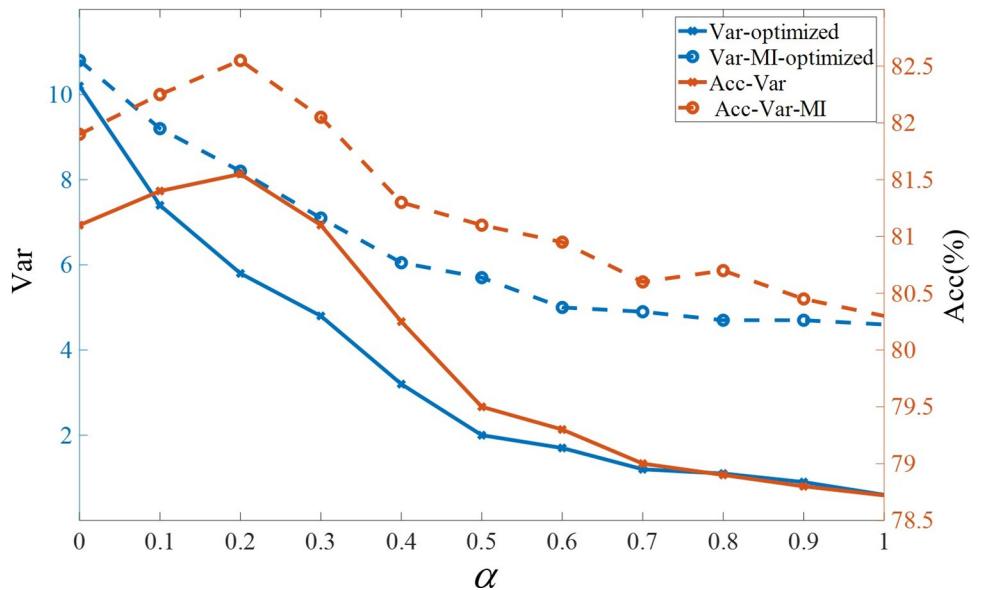


Fig 7. Impact of variance penalty and mutual information enhancement on Arousal dimension.

<https://doi.org/10.1371/journal.pone.0311541.g007>

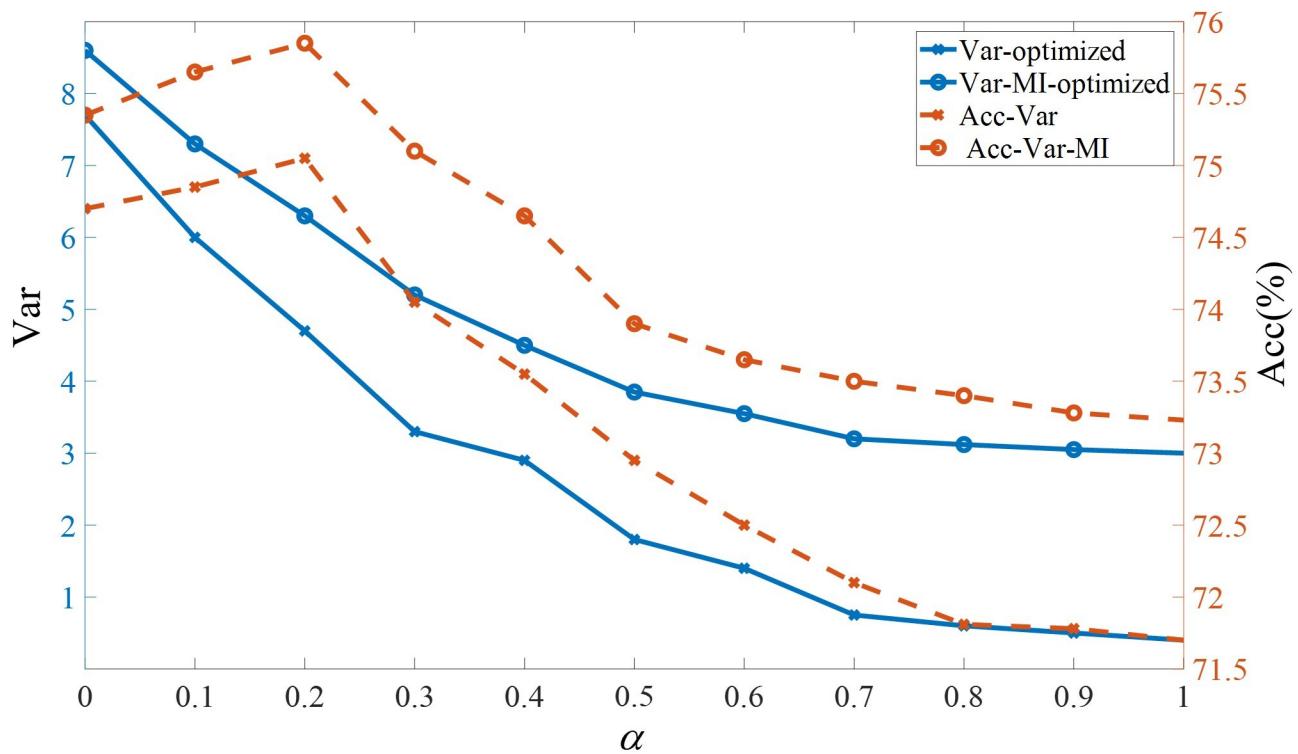


Fig 8. Impact of variance penalty and mutual information enhancement on Valence dimension.

<https://doi.org/10.1371/journal.pone.0311541.g008>

from capturing the overall data distribution and resulting in information loss. Therefore, based on experimental results, a α value of 0.2 is chosen to optimize model performance. Furthermore, the red lines in Figs 7 and 8 demonstrate the performance changes after enhancing mutual information on top of the variance penalty. With the same weight control, optimizing mutual information results in higher variance and accuracy, and at lower α values, mutual information enhancement compensates for insufficient variance penalty. Fig 9 shows that

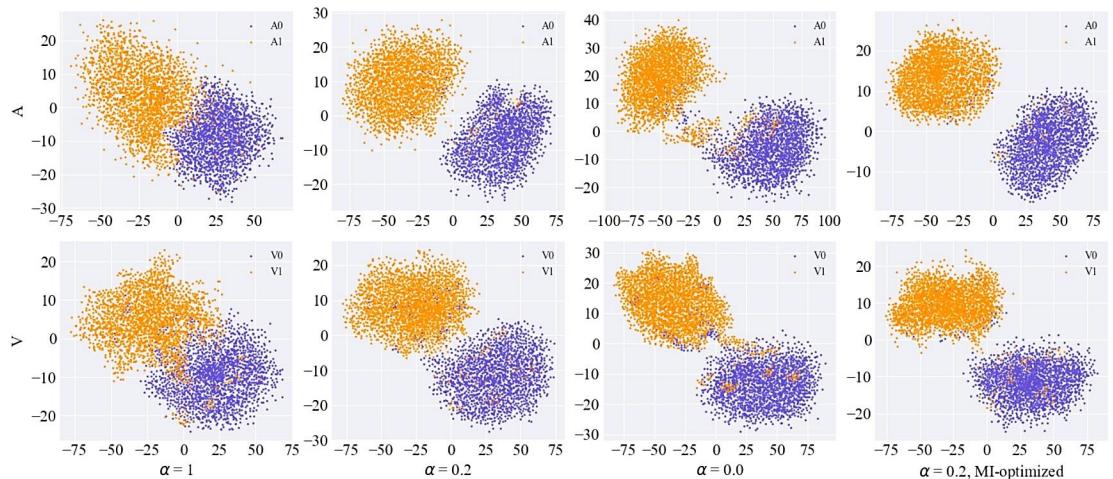


Fig 9. Visualization of latent space under the influence of variance penalty and mutual information enhancement.

<https://doi.org/10.1371/journal.pone.0311541.g009>

after enhancing mutual information, the distances between different clusters become more pronounced, and within-cluster distances become more compact.

4.4.3 Performance analysis of feature disentanglement. For emotion-based music generation, we incorporated a disentanglement mechanism into the CDGMVAE model, learning the emotional representations of rhythm and key features in the Arousal and Valence dimensions, respectively. This section evaluates the impact of changing feature representations on the generated music. Ideally, the two learned feature representations should be independent. For instance, altering the rhythm feature representation should change the rhythm pattern of the generated music, while the key feature remains unchanged. To validate the effectiveness of disentanglement learning, several comparison models were constructed and evaluated under the same experimental settings:

1. EC-VAE [41]: This model uses a bidirectional GRU-based VAE to disentangle pitch and rhythm representations from a single encoder's overall latent variable, and then reconstructs rhythm features and the original input using rhythm and global decoders, respectively.
2. GAN-CVAE [42]: This model defines a latent space where the latent representation is independent of feature values. It learns a generalized latent variable representation devoid of any feature information through adversarial mechanisms, with each controllable feature divided into different categories. These categories are used as conditions to generate music with specific features, and the model is also built using GRU.
3. MuseMorphose [43]: Similar to GAN-CVAE, this model uses Transformer for the main network, focusing more on overall dependency information. It also explores different ways to inject latent variables and feature information into the decoder, effectively utilizing known conditional information.
4. Vanilla CDGMVAE: A simplified version of our model, inspired by the design in [44], which includes only the GMVAE evidence lower bound without additional loss function constraints.
5. CDLSTM-GMVAE: A variant of our model using bidirectional long short-term memory (BiLSTM) networks to construct the GMVAE model.

For feature controllability evaluation, we performed experiments where pairs of samples from the test set were input into the model's encoder. Each pair of samples, denoted as A and B, had their latent variable representations for one feature swapped while keeping the other feature unchanged. The generated music is then compared with the original samples in terms of feature similarity. The similarity is assessed by calculating the cosine similarity between the generated samples and A, B after feature swapping, as defined in Eq (22).

$$sim(a, b) = \langle a, b \rangle / \|a\| \|b\| \quad (22)$$

where a and b denote the feature vectors of generated and original samples, respectively, while $\langle \cdot, \cdot \rangle$ represents the dot product, and $\|\cdot\|$ signifies the norm of the feature vector.

Figs 10 and 11 display bar charts illustrating the variation in feature similarity between the generated samples and original samples A and B after exchanging rhythmic and melodic features, respectively. It is evident from the figures that the CDLSTM-GMVAE and CDGMVAE models effectively disentangle and separate features. Consequently, altering the latent variables of one feature does not significantly impact another feature. When substituting the latent variables of sample A with those of sample B, the generated music closely mimics the rhythmic and melodic patterns of sample B, resulting in higher feature similarity values. Conversely,

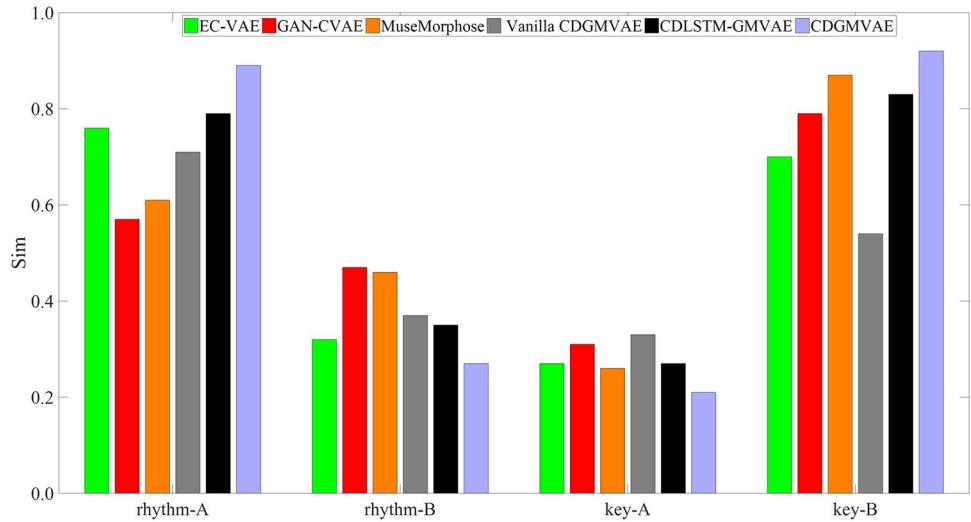


Fig 10. Feature similarity between generated samples and original samples after exchanging rhythmic feature representations.

<https://doi.org/10.1371/journal.pone.0311541.g010>

when proper feature constraints are absent, such as in the Vanilla CDGMVAE and EC-VAE models, although the similarity between features of modified sample A and sample B is high, changes in one feature affect the other, leading to a reduced similarity with the original sample A. This indicates that the rhythmic and melodic feature representations have not been effectively disentangled.

Additionally, for the GAN-CVAE and MuseMorphose models, the effect of feature control is less pronounced compared to other models. The feature similarity with original sample A remains high, while similarity with substituted sample B is lower. This suggests that defining

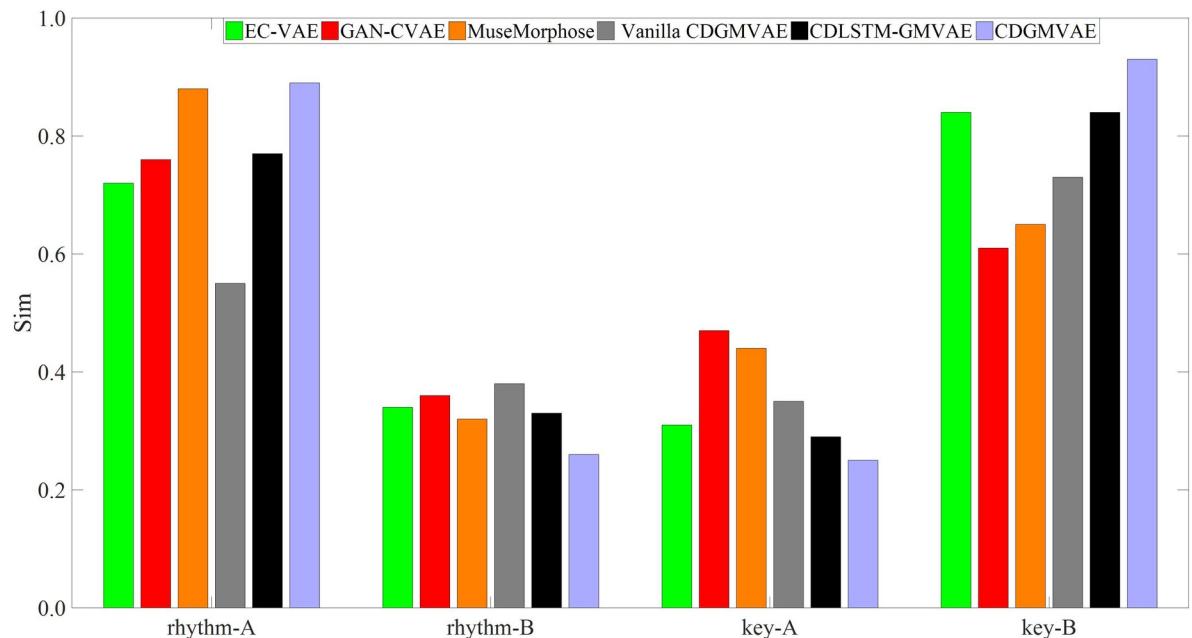


Fig 11. Feature similarity between generated samples and original samples after exchanging melodic feature representations.

<https://doi.org/10.1371/journal.pone.0311541.g011>

unrelated latent spaces for features and segmenting musical features into categorical intervals, while using generalized latent variables and categorical feature labels for music generation control, does not yield satisfactory results. The models tend to learn general data distributions and may overlook subtle feature information. In contrast, the latent space in CDGMVAE directly reflects semantic information related to features. Using latent variables representing specific features provides more detailed information, allowing the generation model to control the output more precisely. Thus, for emotion-driven music generation models controlled by features, it is crucial not only to have effective feature representations but also to ensure that these representations are independent and do not interfere with one another.

Through the above experimental analysis, the proposed emotion-driven music generation model, CDGMVAE, successfully disentangles rhythmic and melodic feature representations related to emotions from the original music sequences. It effectively manipulates these feature representations to achieve desired emotional variations in the Arousal and Valence dimensions. (More experiment details can be seen in the supplementary material [S1 Text])

4.4.4 Interpolative emotion transformation. In the previous sections, the disentangled control of rhythmic and melodic features for music emotion transformation is assessed by exchanging features between two pieces of music. To fully leverage the discrete latent space of CDGMVAE, we employed a feature disentanglement mechanism combined with interpolation methods to enable mutual transformation of musical emotions by mapping the latent variables of the current music to the latent space of the target emotion. When transforming between high and low dimensions of a specific emotional dimension, the difference between the Gaussian component means $\mu_{i,\text{target}}$ in the target emotion space and the Gaussian component means $\mu_{i,\text{source}}$ in the current emotion space is first computed. This difference is then added to the latent variable $z_{i,\text{source}}$ of the current emotion to obtain the latent variable $z_{i,\text{target}}$ corresponding to the target emotion. Finally, this latent variable is input into the CDGMVAE decoder to generate a new sample. The entire process is described by Eq (23):

$$z_{i,\text{target}} = z_{i,\text{source}} + \lambda \cdot (\mu_{i,\text{target}} - \mu_{i,\text{source}}) \quad (23)$$

where i denotes the emotion dimension to be transformed, and the parameter $\lambda \in [0,1]$ controls the degree of closeness between the current and target emotions. For clearer visualization of emotion changes in this experiment, λ is set to 1.

Using the aforementioned method, we conducted experiments on the test set to assess music emotion conversion. Based on high and low-dimensional emotion spaces composed of Arousal and Valence, we performed conversions among four emotions: happy, tense, sad, and calm. The final results were evaluated for accuracy using the music emotion classification model EMPOIA, as shown in the confusion matrix in [Table 4](#).

Overall, transforming latent variables to target emotion clusters using interpolation achieves a certain degree of emotion conversion. This indicates that the Gaussian components for each emotion are well-separated and that each emotion is accurately represented through latent variable information. However, the overall prediction accuracy is not very high,

Table 4. Evaluation of the accuracy of music emotion transfer.

	T-Happiness	T-Tension	T-Sadness	T-Calmness
S-Happiness	-	60.84%	59.70%	71.56%
S-Tension	63.45%	-	67.81%	57.63%
S-Sadness	58.29%	65.48%	-	64.80%
S-Calmness	70.26%	56.21%	62.35%	-

<https://doi.org/10.1371/journal.pone.0311541.t004>

influenced by the EMPOIA classification model, which operates on four emotion categories rather than predicting high and low dimensions of individual emotion dimensions. The experimental results also show that the accuracy of emotion conversion in the Arousal dimension is higher compared to the Valence dimension, with the lowest accuracy observed for conversions across both dimensions. This is consistent with the conclusions in Section 4.4.1, indicating that music is more easily distinguished in the Arousal dimension than in the Valence dimension, and conversions across both dimensions are more challenging. Among the four emotions, the conversion accuracy between happy and calm is the highest, suggesting a high similarity between these two emotions, which allows for effective conversion through changes in rhythmic patterns alone.

5. Conclusion

This paper presents the Emotion-Driven Music Generation Model, CDGMVAE, which utilizes GMVAE for semi-supervised clustering inference during training. Compared to fully supervised generative models, CDGMVAE can learn richer data distributions from a small amount of labeled data, effectively mitigating the issue of insufficient emotional music data and enhancing the model's ability to infer and generate different emotional music categories. To address the mode collapse problem inherent in GMVAE, we analyzed the evidence lower bound and identified the variance regularization term and mutual information suppression term as key contributors. Therefore, we introduced penalties and enhancements for these factors. Experimental results demonstrate that this approach ensures better separation of different emotions in the latent space, strengthens the correlation between music and emotional information, and improves the robustness and generalization of the semi-supervised model. Given that existing music emotion generation models lack interpretability of emotions, we propose establishing connections between emotions and musical rhythmic and melodic features. By introducing a feature disentanglement mechanism to learn emotional representations of these features and incorporating adversarial loss to enhance feature disentanglement, we achieve controlled manipulation of music emotion transformations. Additionally, we employed Transformer-XL as the encoder and decoder for GMVAE, which effectively learns longer contextual dependencies in music sequences, further enhancing the realism of the generated music.

Supporting information

S1 Text. Supplementary experimental results.
(DOCX)

Author Contributions

Conceptualization: Jie Pan.

Software: Xiao Han.

Writing – original draft: Zihao Ning.

Writing – review & editing: Jie Pan.

References

1. Liu Z, Xu W, Zhang W, et al. An emotion-based personalized music recommendation framework for emotion improvement[J]. *Information Processing & Management*, 2023, 60(3): 103256.
2. Ji S, Yang X, Luo J. A survey on deep learning for symbolic music generation: Representations, algorithms, evaluations, and challenges[J]. *ACM Computing Surveys*, 2023, 56(1): 1–39.

3. Wang X, Jiang H, Mu M, Dong Y. A trackable multi-domain collaborative generative adversarial network for rotating machinery fault diagnosis [J].
4. Luo P, Yin Z, Yuan D, et al. A Novel Generative Adversarial Networks via Music Theory Knowledge for Early Fault Intelligent Diagnosis of Motor Bearings[J]. *IEEE Transactions on Industrial Electronics*, 2023.
5. Hadjeres G, Nielsen F. Anticipation-RNN: Enforcing unary constraints in sequence generation, with application to interactive music generation[J]. *Neural Computing and Applications*, 2020, 32(4): 995–1005.
6. Zhao K, Li S, Cai J, et al. An emotional symbolic music generation system based on LSTM networks [C]//2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). IEEE, 2019: 2039–2043.
7. Johnson D D. Generating polyphonic music using tied parallel networks[C]//International conference on evolutionary and biologically inspired music and art. Cham: Springer International Publishing, 2017: 128–143.
8. Ferreira L N, Whitehead J. Learning to generate music with sentiment[J]. arXiv preprint arXiv:2103.06125, 2021.
9. Hung H T, Ching J, Doh S, et al. EMOPIA: A multi-modal pop piano dataset for emotion recognition and emotion-based music generation[J]. arXiv preprint arXiv:2108.01374, 2021.
10. Sulun S, Davies M E P, Viana P. Symbolic music generation conditioned on continuous-valued emotions[J]. *IEEE Access*, 2022, 10: 44617–44626.
11. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks[J]. *Communications of the ACM*, 2020, 63(11): 139–144. <https://doi.org/10.1145/3373453.3394370>
12. Wang X, Jiang H, Wu Z, Yang Q. Adaptive variational autoencoding generative adversarial networks for rolling bearing fault diagnosis[J]. *Advanced Engineering Informatics*, 2023, 56: 102027. <https://doi.org/10.1016/j.aei.2023.102027>
13. Grekow J, Dimitrova-Grekow T. Monophonic music generation with a given emotion using conditional variational autoencoder[J]. *IEEE Access*, 2021, 9: 129088–129101.
14. Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder–decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.
15. Huang C F, Huang C Y. Emotion-based AI music generation system with CVAE-GAN[C]//2020 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE). IEEE, 2020: 220–222.
16. Tan H H. ChordAL: A Chord-Based Approach for Music Generation using Bi-LSTMs[C]//ICCC. 2019: 364–365.
17. Majidi M, Toroghi R M. A combination of multi-objective genetic algorithm and deep learning for music harmony generation[J]. *Multimedia Tools and Applications*, 2023, 82(2): 2419–2435.
18. Chen K, Wang C, Berg-Kirkpatrick T, et al. Music sketchnet: Controllable music generation via factorized representations of pitch and rhythm[J]. arXiv preprint arXiv:2008.01291, 2020.
19. Luo Y J, Agres K, Herremans D. Learning disentangled representations of timbre and pitch for musical instrument sounds using gaussian mixture variational autoencoders[J]. arXiv preprint arXiv:1906.08152, 2019.
20. Tan H H, Herremans D. Music fadernets: Controllable music generation based on high-level features via low-level feature modelling[J]. arXiv preprint arXiv:2007.15474, 2020.
21. Wu S L, Yang Y H. MuseMorphose: Full-song and fine-grained piano music style transfer with one transformer VAE[J]. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023, 31: 1953–1967.
22. BRUNNER G, KONRAD A, WANG Y, et al. MIDI-VAE: Modeling dynamics and instrumentation of music with applications to style transfer[C]//Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR). 2018: 747–754.
23. Dai Z, Yang Z, Yang Y, et al. Transformer-XL: Attentive Language Models beyond a Fixed-Length Context[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL). 2019: 2978–2988.
24. Li Z, Zhao Y, Xu H, et al. Unsupervised clustering through gaussian mixture variational autoencoder with non-reparameterized variational inference and std annealing[C]//2020 International Joint Conference on Neural Networks (IJCNN). IEEE, 2020: 1–8.
25. Jiang Z, Zheng Y, Tan H, et al. Variational deep embedding: An unsupervised and generative approach to clustering[J]. arXiv preprint arXiv:1611.05148, 2016.
26. Zhao T, Lee K, Eskenazi M. Unsupervised discrete sentence representation learning for interpretable neural dialog generation[J]. arXiv preprint arXiv:1804.08069, 2018.

27. Rezaabad A L, Vishwanath S. Learning representations by maximizing mutual information in variational autoencoders[C]//2020 IEEE International Symposium on Information Theory (ISIT). IEEE, 2020: 2729–2734.
28. Vaswani A. Attention is all you need[J]. arXiv preprint arXiv:1706.03762, 2017.
29. Jiang J, Xia G G, Carlton D B, et al. Transformer vae: A hierarchical model for structure-aware and interpretable music representation learning[C]//ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020: 516–520.
30. Wu S L, Yang Y H. MuseMorphose: Full-song and fine-grained piano music style transfer with one transformer VAE[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2023, 31: 1953–1967.
31. Dong H W, Hsiao W Y, Yang L C, et al. Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2018, 32(1).
32. Bertin-Mahieux T, Ellis D P W, Whitman B, et al. The million song dataset[J]. 2011.
33. Huang Y S, Yang Y H. Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions[C]//Proceedings of the 28th ACM international conference on multimedia. 2020: 1180–1188.
34. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks[C]//Proceedings of the thirteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 2010: 249–256.
35. Hung H T, Ching J, Doh S, et al. EMOPIA: A multi-modal pop piano dataset for emotion recognition and emotion-based music generation[J]. arXiv preprint arXiv:2108.01374, 2021.
36. Zheng K, Meng R, Zheng C, et al. EmotionBox: A music-element-driven emotional music generation system based on music psychology[J]. Frontiers in Psychology, 2022, 13: 841926.
37. Grekow J, Dimitrova-Grekow T. Monophonic music generation with a given emotion using conditional variational autoencoder[J]. IEEE Access, 2021, 9: 129088–129101.
38. Kingma D P, Mohamed S, Jimenez Rezende D, et al. Semi-supervised learning with deep generative models[J]. Advances in neural information processing systems, 2014, 27.
39. Joy T, Schmon S M, Torr P H S, et al. Capturing label characteristics in vaes[J]. arXiv preprint arXiv:2006.10102, 2020.
40. Tan H H, Herremans D. Music fadernets: Controllable music generation based on high-level features via low-level feature modelling[J]. arXiv preprint arXiv:2007.15474, 2020.
41. Yang R, Wang D, Wang Z, et al. Deep music analogy via latent representation disentanglement[J]. arXiv preprint arXiv:1906.03626, 2019.
42. Kawai L, Esling P, Harada T. Attributes-Aware Deep Music Transformation[C]//ISMIR. 2020: 670–677.
43. Wu S L, Yang Y H. MuseMorphose: Full-song and fine-grained piano music style transfer with one transformer VAE[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2023, 31: 1953–1967.
44. Luo Y J, Agres K, Herremans D. Learning disentangled representations of timbre and pitch for musical instrument sounds using gaussian mixture variational autoencoders[J]. arXiv preprint arXiv:1906.08152, 2019.