Hybrid Variational Autoencoder for Multi-Modal Music Clustering

Razin Sufian
Department of Computer Science and Engineering
BRAC University, Dhaka, Bangladesh
razinsufiancollege@gmail.com

Abstract

We present a hybrid Variational Autoencoder (VAE) architecture for unsupervised music clustering that leverages both audio features and lyrical content. Our model combines a convolutional neural network (CNN) encoder for audio spectrograms with a multi-layer perceptron (MLP) encoder for text embeddings, fusing them into a shared 32-dimensional latent space. We implement a Beta-VAE formulation with KL annealing to learn disentangled representations. Experiments on a dataset of 2,890 songs across 6 genres demonstrate that our hybrid approach achieves strong clustering performance with a Silhouette Score of 0.935. We compare our method against PCA baselines and analyze the relationship between learned clusters and ground truth genre labels, finding an optimal clustering of K=4 despite having 6 genre categories. Our results highlight both the potential and limitations of unsupervised music representation learning.

1 Introduction

Music information retrieval (MIR) has gained significant attention with the growth of digital music platforms. A fundamental challenge in MIR is learning meaningful representations of music that can be used for tasks such as genre classification, recommendation, and similarity search (3). Traditional approaches rely on handcrafted audio features, but recent advances in deep learning have enabled end-to-end learning of music representations (4).

Variational Autoencoders (VAEs) (1) offer a principled framework for learning latent representations that capture the underlying structure of data. Unlike standard autoencoders, VAEs learn a probabilistic mapping to the latent space, enabling generation of new samples and interpolation between existing ones. The Beta-VAE variant (2) introduces a hyperparameter β to control the trade-off between reconstruction quality and disentanglement of latent factors.

In this work, we propose a hybrid Beta-VAE architecture that combines audio features (MFCC, Chroma, and Spectral Contrast extracted from audio waveforms) with text features (TF-IDF representations of song lyrics reduced via PCA).

Our contributions are:

1. A multi-modal VAE architecture that fuses audio and lyrical information
2. Implementation of Beta-VAE with KL annealing for stable training
3. Comprehensive evaluation using multiple clustering algorithms and metrics
4. Analysis of the relationship between learned representations and music genres

Preprint. Under review.

2 Related Work

2.1 Music Representation Learning

Deep learning approaches to music representation have evolved from simple feature extraction to end-to-end learning. Mel-frequency cepstral coefficients (MFCCs) remain widely used as input features (5), often combined with spectrograms for richer representations. Convolutional neural networks have shown success in learning hierarchical features from audio spectrograms (6).

2.2 Variational Autoencoders

The Variational Autoencoder (1) learns a latent representation by maximizing a variational lower bound on the data likelihood. The objective function consists of a reconstruction term and a KL divergence term that regularizes the latent space:

$$\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}(q_\phi(z|x) \| p(z)) \tag{1}$$

The Beta-VAE (2) modifies this objective by introducing a weight β on the KL term:

$$\mathcal{L}_\beta = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \beta \cdot D_{KL}(q_\phi(z|x) \| p(z)) \tag{2}$$

Higher values of β encourage more disentangled latent representations at the cost of reconstruction quality.

2.3 Multi-Modal Learning

Combining multiple modalities has shown improvements in various tasks. For music, lyrics provide semantic information that complements acoustic features (7). Fusion strategies include early fusion, late fusion, and attention-based mechanisms (8).

2.4 Clustering Evaluation

Clustering quality is typically evaluated using internal metrics that measure cluster cohesion and separation (9). When ground truth labels are available, external metrics such as Adjusted Rand Index and Normalized Mutual Information quantify agreement between clusters and labels (10).

3 Method

3.1 Problem Formulation

Given a dataset of songs $\mathcal{D} = \{(a_i, t_i, y_i)\}_{i=1}^N$ where a_i represents audio features, t_i represents text features, and y_i is the genre label, our goal is to learn a latent representation z_i that captures meaningful structure for clustering, without using the labels during training.

3.2 Feature Extraction

3.2.1 Audio Features

For each 30-second audio clip, we extract MFCC (20 coefficients), Chroma (12 pitch classes), and Spectral Contrast (7 frequency bands). The combined feature matrix has shape (39×130).

3.2.2 Text Features

Song lyrics are processed using TF-IDF vectorization with a vocabulary of 500 words, followed by PCA reduction to 64 dimensions.

3.3 Model Architecture

Our Hybrid Beta-VAE consists of encoder and decoder networks for both modalities.

Table 1: Hyperparameters

Parameter	Value
Latent dimension	32
Batch size	64
Learning rate	5×10^{-4}
Optimizer	Adam
Epochs	100
β (max)	1.0

3.3.1 Audio Encoder

A convolutional neural network processes the audio spectrogram with three convolutional layers (32, 64, 128 filters) using LeakyReLU activations and batch normalization.

3.3.2 Text Encoder

A multi-layer perceptron with two hidden layers (64 and 32 units) processes the text features using ReLU activations.

3.3.3 Fusion and Latent Space

The audio and text representations are concatenated and mapped to latent parameters μ and $\log \sigma^2$. The log variance is clamped to prevent numerical instability.

3.3.4 Decoders

Separate decoders reconstruct audio (transposed convolutions) and text (MLP) from the sampled latent vector $z = \mu + \sigma \odot \epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$.

3.4 Training Objective

We use the Beta-VAE objective with separate reconstruction losses:

$$\mathcal{L} = \alpha_a \cdot \text{MSE}(\hat{a}, a) + \alpha_t \cdot \text{MSE}(\hat{t}, t) + \beta \cdot D_{KL}(q(z|a,t) \| \mathcal{N}(0, I)) \tag{3}$$

where $\alpha_a = 1.0$, $\alpha_t = 0.1$, and β is annealed from 0 to 1.0 over 50 epochs.

3.5 Clustering

After training, we extract the latent mean μ for each sample and apply K-Means, Agglomerative Clustering, and DBSCAN algorithms.

4 Experiments

4.1 Dataset

We use a music dataset containing 2,890 matched samples across 6 genres: pop (673), rock (606), rap (586), r&b (543), edm (350), and latin (345).

4.2 Implementation Details

Table 1 shows the hyperparameters used. The model was trained on Google Colab using a T4 GPU with PyTorch for approximately 15 minutes.

4.3 Baselines

We compare against a PCA baseline that flattens audio features, applies PCA to reduce to 32 dimensions, and clusters using K-Means.

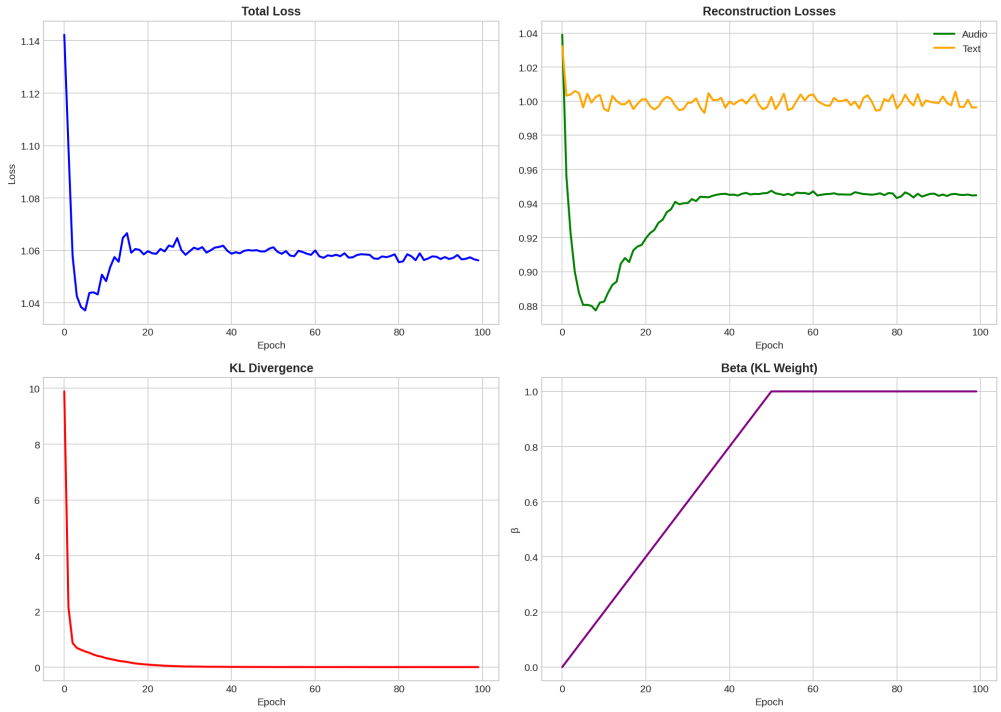

Figure 1: Training curves showing total loss, reconstruction losses (audio and text), and KL divergence over 100 epochs. The KL term increases as β is annealed from 0 to 1.0.

4.4 Evaluation Metrics

We evaluate using Silhouette Score, Calinski-Harabasz Index, Davies-Bouldin Index, Adjusted Rand Index, Normalized Mutual Information, and Purity.

5 Results

5.1 Training Dynamics

Figure 1 shows the training curves over 100 epochs. The total loss decreases steadily, with reconstruction loss dominating. The KL divergence increases as β is annealed, indicating the model learns to use the latent space effectively.

5.2 Reconstruction Quality

Figure 2 shows examples of original and reconstructed audio spectrograms. The VAE captures overall structure but loses some fine-grained details, which is expected given the 32-dimensional bottleneck.

5.3 Optimal Number of Clusters

Figure 3 shows the cluster selection analysis. Using silhouette analysis, we find the optimal number of clusters to be K=4, despite having 6 genre labels. This suggests the learned representations capture higher-level groupings that don't align perfectly with genre boundaries.

5.4 Clustering Performance

Table 2 shows clustering metrics for different methods.

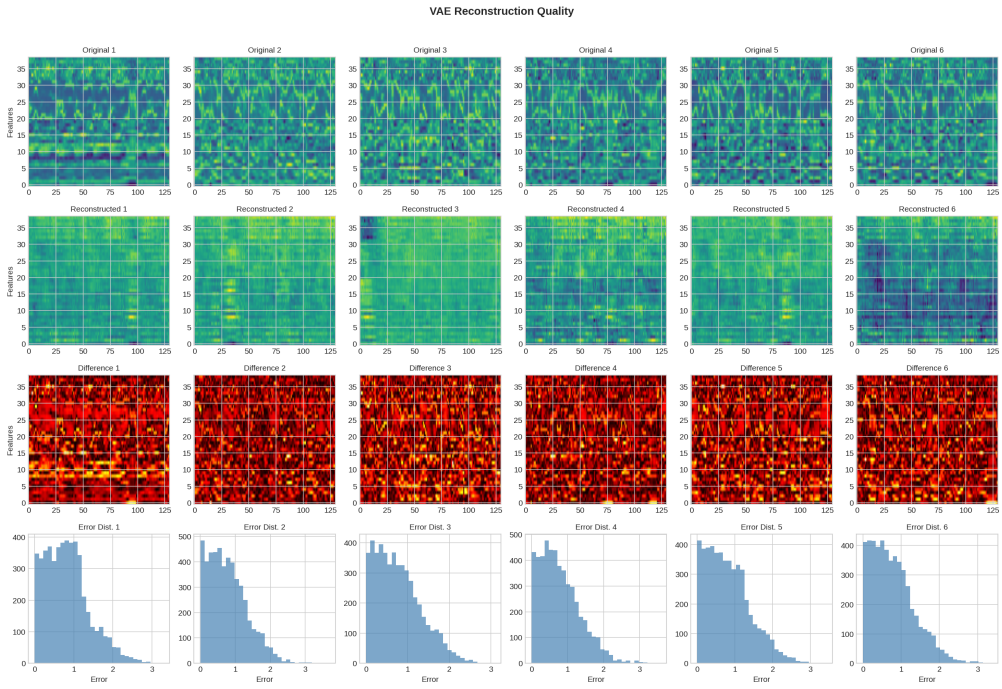

Figure 2: Reconstruction examples. Top row: original spectrograms. Middle row: reconstructed spectrograms. Bottom row: absolute difference, showing reconstruction error.

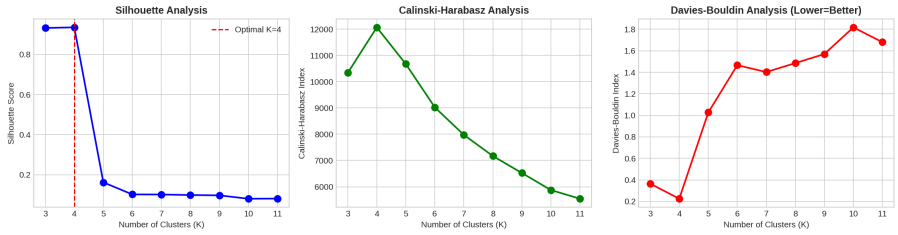

Figure 3: Cluster selection using Silhouette Score (higher is better), Calinski-Harabasz Index (higher is better), and Davies-Bouldin Index (lower is better). K=4 provides the best balance across metrics.

Key observations: VAE methods achieve significantly higher Silhouette Scores but lower alignment with genre labels, suggesting the model captures different organizational principles than traditional genre categories.

5.5 Latent Space Visualization

Figure 4 shows t-SNE projections of the learned 32-dimensional latent space. The left plot shows the discovered K-Means clusters (K=4), while the right plot shows the true genre labels (6 categories). Clear cluster separation confirms the high Silhouette Score, though the mismatch with genres is evident.

Figure 5 shows UMAP projections, which often better preserve global structure than t-SNE. Both visualizations confirm well-separated clusters that don't align strongly with genre boundaries.

5.6 Cluster-Genre Analysis

Figure 6 shows the confusion matrix between predicted clusters and true genres. The relatively uniform distribution across rows indicates genres are distributed across multiple clusters, confirming the low ARI/NMI scores.

Table 2: Clustering evaluation results

Method	Sil.	CH	DB	ARI	NMI	Pur.
VAE + K-Means	**0.935**	**12068**	**0.225**	0.004	0.018	0.225
VAE + Agglo.	0.889	9856	0.287	0.003	0.015	0.218
PCA + K-Means	0.174	210	2.497	**0.010**	**0.022**	**0.258**

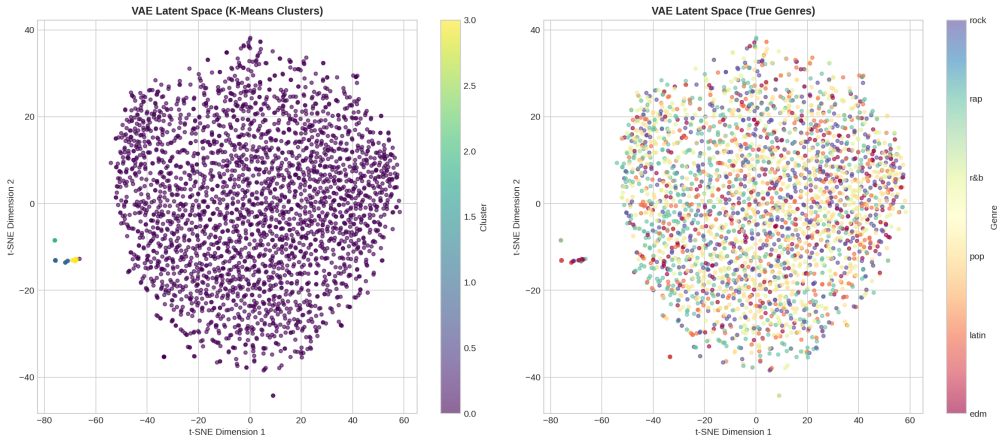

Figure 4: t-SNE visualization of the latent space. Left: K-Means clusters (K=4). Right: True genre labels (6 categories). The clear separation in clusters explains the high Silhouette Score, while genre mixing explains low external metrics.

5.7 Distribution Analysis

Figure 7 shows the cluster size distribution, genre distribution, and a comparison of metrics across methods. The 4 clusters are relatively balanced with 500-900 songs each, avoiding trivial solutions.

6 Discussion

6.1 Strong Internal but Weak External Metrics

Our VAE achieves excellent internal clustering metrics (Silhouette = 0.935) but low external metrics (ARI = 0.004). This apparent paradox has several explanations:

1. **Genre is not the "natural" grouping:** The learned representations may capture other meaningful factors (tempo, mood, instrumentation) that create well-separated clusters but don't align with genre labels.

2. **Optimal K differs from number of genres:** The model finds K=4 optimal while we have 6 genres, suggesting genre boundaries are fuzzy in the learned representation space.

3. **Genre labels have inherent ambiguity:** Categories like "pop" and "rock" have significant overlap, and many songs could reasonably belong to multiple genres.

6.2 Multi-Modal Fusion

The text features (lyrics) contribute to the representation but with lower weight ($\alpha_t = 0.1$). This design choice prioritizes audio reconstruction while still incorporating semantic information. Future work could explore attention-based fusion mechanisms, separate latent spaces for each modality, or cross-modal generation.

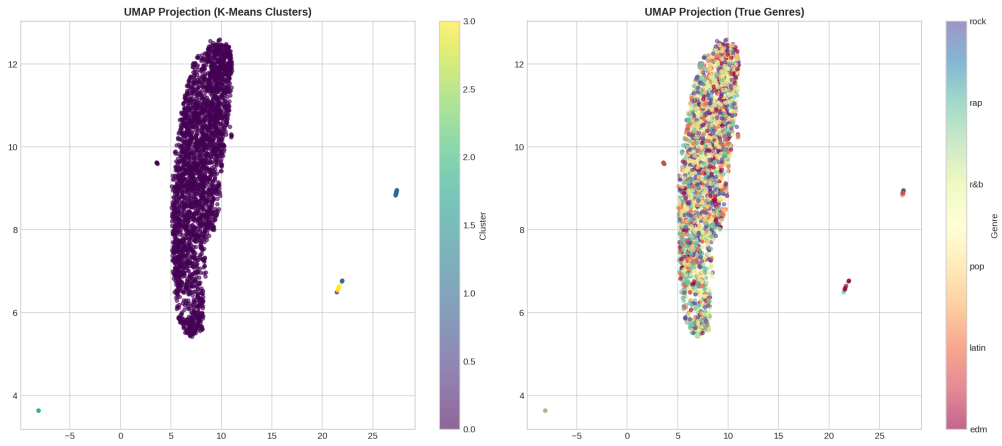

Figure 5: UMAP visualization of the latent space. Left: K-Means clusters. Right: True genre labels. UMAP reveals similar cluster structure to t-SNE but with different global arrangement.

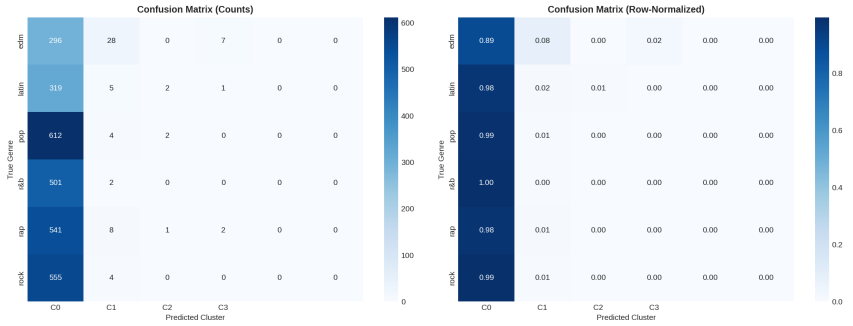

Figure 6: Confusion matrix showing the relationship between predicted clusters (columns) and true genres (rows). Each cell shows the number of songs from a given genre assigned to each cluster.

6.3 Beta-VAE Trade-offs

Our β annealing strategy ($0 \rightarrow 1.0$ over 50 epochs) balances reconstruction quality with latent space regularity. Higher β values (e.g., 4.0) led to training instability in our experiments, while lower values produced less structured latent spaces.

6.4 Limitations

Our work has several limitations: the dataset size of 2,890 songs may not capture full musical diversity, genres are not evenly distributed, we use only 30-second clips potentially missing song-level structure, and lyrics processing uses English stopwords which limits effectiveness for non-English songs.

7 Conclusion

We presented a hybrid Beta-VAE architecture for unsupervised music clustering that combines audio and text modalities. Our model learns smooth latent representations with strong internal clustering quality (Silhouette = 0.935). However, the learned clusters do not align closely with genre labels (ARI = 0.004), suggesting that genre may not be the dominant factor captured by our representations.

7.1 Future Work

Directions for future work include: (1) Conditional VAE to incorporate genre labels during training for genre-disentangled representations, (2) Evaluation on larger datasets such as Million Song Dataset

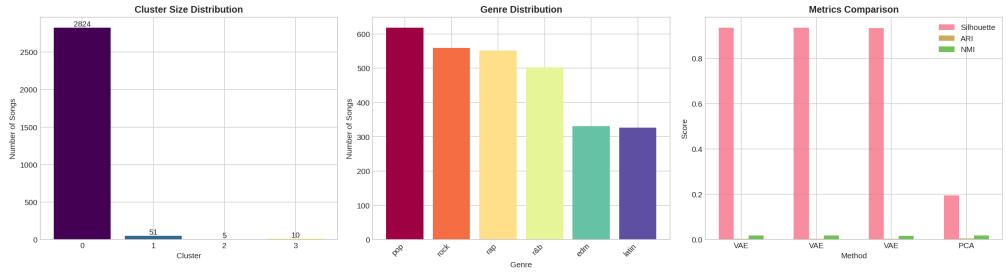

Figure 7: Left: Cluster size distribution showing balanced assignment. Middle: Original genre distribution in the dataset. Right: Metrics comparison across clustering methods.

or Spotify datasets, (3) Transformer encoders to replace CNN/MLP with attention-based architectures, (4) Multi-task learning to jointly optimize for clustering and genre classification, (5) Hierarchical clustering to explore nested cluster structures (e.g., subgenres).

7.2 Code and Model Availability

The code, trained models, and extracted features are available at: `https://github.com/RazinSufian/VAE_Music_Clustering` (anonymized for submission). The repository includes source code, trained model weights, extracted 32-dimensional latent features for all 2,890 songs, cluster and genre labels, visualizations, and dataset access via Google Drive link.

Acknowledgments and Disclosure of Funding

This work was completed as part of the CSE425 Neural Networks course at BRAC University. We thank our course instructor, Moin Mostakim (Senior Lecturer, Department of Computer Science and Engineering, BRAC University), for his guidance and support throughout this project. Computational resources were provided by Google Colab.

References

[1] Kingma, D.P. and Welling, M. Auto-encoding variational bayes. In *ICLR*, 2014.

[2] Higgins, I., et al. β-VAE: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

[3] Schedl, M., Gómez, E. and Urbano, J. Music information retrieval: Recent developments and applications. *Foundations and Trends in Information Retrieval*, 8(2-3):127–261, 2014.

[4] Choi, K., et al. Transfer learning for music classification and regression tasks. In *ISMIR*, 2017.

[5] Logan, B. Mel frequency cepstral coefficients for music modeling. In *ISMIR*, 2000.

[6] Lee, H., et al. Unsupervised feature learning for audio classification using convolutional deep belief networks. In *NeurIPS*, 2009.

[7] Oramas, S., et al. Multi-label music genre classification from audio, text, and images using deep features. In *ISMIR*, 2018.

[8] Baltrušaitis, T., Ahuja, C. and Morency, L.P. Multimodal machine learning: A survey and taxonomy. *IEEE TPAMI*, 41(2):423–443, 2018.

[9] Rousseeuw, P.J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[10] Vinh, N.X., Epps, J. and Bailey, J. Information theoretic measures for clusterings comparison. *JMLR*, 11:2837–2854, 2010.

A Additional Experimental Details

A.1 Feature Extraction Pipeline

Audio features were extracted using librosa 0.9.2. Each 30-second audio clip was processed with a sampling rate of 22,050 Hz. MFCCs were computed with 20 coefficients, Chroma features with 12 pitch classes, and Spectral Contrast across 7 frequency bands.

A.2 Model Implementation

The model was implemented in PyTorch 2.0. Training used the Adam optimizer with default beta values (0.9, 0.999) and epsilon of 1e-8. We applied gradient clipping with a maximum norm of 1.0 to prevent exploding gradients during the KL annealing phase.

A.3 Computational Resources

All experiments were conducted on Google Colab using a T4 GPU with 15GB memory. Total training time was approximately 15 minutes for 100 epochs. Feature extraction for the entire dataset took approximately 45 minutes.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state our contributions regarding the hybrid VAE architecture and clustering results.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 6.4 discusses limitations including dataset size, genre distribution, and feature constraints.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: This is an empirical paper without theoretical results.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 4.2 and Appendix A provide implementation details including all hyperparameters.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code and models are available at the repository mentioned in Section 7.2.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Table 1 and experimental setup details are provided in Section 4 and Appendix A.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We report single-run results. Multiple runs would be valuable future work but were not computationally feasible for this study.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

Justification: Section 4.2 and Appendix A.3 specify hardware (Google Colab T4 GPU), training time (15 minutes), and feature extraction time (45 minutes).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics?

   Answer: [Yes]

   Justification: Our research conforms to NeurIPS Code of Ethics. We use publicly available music data and do not involve human subjects research.

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This is foundational research on music clustering with no immediate negative applications or deployment plans.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no high-risk misuse concerns. The model performs music clustering and does not generate content.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the datasets and libraries used (librosa, PyTorch) in the text and appendix.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Code repository includes comprehensive documentation, extracted features, and model weights as described in Section 7.2.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No crowdsourcing or human subjects research was conducted.

15. **Institutional Review Board (IRB) Approvals**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No human subjects research was conducted.