Variational Autoencoders for Multi-Modal Hybrid-Language Music Clustering: A Systematic Comparison

Moin Mostakim
Department of Computer Science
Neural Networks Course
Submission Date: January 10, 2026
moin.mostakim@example.edu

January 2, 2026

Abstract

Unsupervised music clustering across multiple languages remains challenging due to acoustic similarity across linguistic boundaries and fluid genre definitions. We present a systematic empirical study comparing seven Variational Autoencoder (VAE) architectures on a custom-curated hybrid-language music dataset. We assembled and balanced 180 full-length songs (45 per language) across 4 languages (Arabic, English, Hindi, Spanish) and 3 genres (Pop, Rock, Hip-Hop), manually pairing all 180 songs with their corresponding lyrics for multimodal experiments. To increase training samples while maintaining temporal independence, we developed a windowing protocol that extracts overlapping 30-second clips (window=30s, hop=20s, 33% overlap) from full-length songs, yielding 2,107 clips. Critically, we implement song-level dataset splitting—partitioning the 180 original songs before windowing into train (80%), validation (10%), and test (10%) sets—to prevent data leakage, ensuring no clips from the same song appear in multiple splits. We systematically evaluate Basic VAE, Convolutional VAE, Beta-VAE, Conditional VAEs (language and genre), VaDE, and Multimodal VAE across three clustering algorithms (K-means, Agglomerative, GMM) using six metrics (NMI, ARI, Silhouette, V-Measure, Calinski-Harabasz, Davies-Bouldin). Results reveal task-specific architectural preferences: for genre clustering, the simplest Basic VAE (5.5M parameters) achieves best performance (mean NMI=0.0815 across methods, max=0.0969 with Agglomerative), outperforming sophisticated architectures including 64.5M-parameter Conv VAE; for language clustering, Multimodal VAE achieves best results (mean NMI=0.0307, max=0.0440 with Agglomerative), demonstrating benefits of audio-lyrics fusion. We discover a reconstruction-clustering disconnect: Multimodal VAE achieves lowest validation loss (0.5502) but poor genre clustering (mean NMI=0.0095), demonstrating that reconstruction quality does not guarantee clustering effectiveness. Task difficulty analysis reveals an NMI ceiling below 0.10 across all models, indicating fundamental dataset-scale limitations. Our methodological contributions include the song-level splitting protocol preventing clip leakage across data splits, comprehensive baseline comparisons (PCA, raw features), and systematic multi-algorithm evaluation revealing that Agglomerative clustering outperforms K-means and GMM. Results demonstrate that for small-scale multilingual music datasets, architectural simplicity and rigorous data splitting outweigh model complexity.

1 Introduction

The automatic organization and clustering of music remains a fundamental challenge in Music Information Retrieval (MIR), with applications spanning recommendation systems, playlist generation, and musicological analysis [29]. While genre classification has traditionally relied on supervised learning with labeled datasets [20], the emergence of multilingual music platforms and hybrid-language content has exposed critical limitations: genre boundaries are increasingly fluid, music from different linguistic backgrounds exhibits acoustic similarity despite cultural differences, and manual annotation scales poorly to millions of tracks [31]. These challenges motivate unsupervised learning approaches that can discover latent structures without explicit labels.

Variational Autoencoders (VAEs) [17] have emerged as powerful frameworks for learning interpretable latent representations from highdimensional data. In music, VAEs enable smooth interpolation between musical styles [28], disentangled representations of musical attributes [12], and joint clustering through probabilistic mixture models [16]. However, existing VAE applications predominantly focus on *generative modeling*—synthesizing new melodies, timbres, or harmonies—rather than *clustering* for music organization. Furthermore, most work targets symbolic music (MIDI) or single-language datasets, leaving hybrid-language music clustering critically underexplored.

1.1 Motivation and Research Questions

Our work addresses three core challenges in multilingual music clustering:

Dataset Scarcity. Existing MIR benchmarks like GTZAN [32] (1,000 clips, English-centric) and MTG-Jamendo [1] (55K tracks, primarily European languages) lack balanced representation across diverse linguistic and cultural contexts. Multilingual datasets like MIR-MLPop [33] remain small (90 tracks) or focus on lyrics without audio-lyrics alignment [14]. There is no established benchmark for hybrid-language music clustering with balanced genre-language distribution.

Data Leakage in Temporal Data. Music machine learning often employs windowing to increase training samples by extracting overlapping clips from full-length songs. However, standard random train/test splitting risks severe data leakage: clips from the same song appear across splits, artificially inflating performance metrics. This methodological pitfall remains inadequately addressed in the literature.

Architecture Selection for Clustering. While sophisticated VAE variants—Convolutional VAEs for spectrograms [9], Beta-VAEs for disentanglement [12], Conditional VAEs for attribute control, VaDE for joint clustering [16], and multimodal fusion for audio-lyrics integration—have proven effective for generation, their comparative effectiveness for *clustering* remains unclear. Does architectural complexity improve cluster quality? Do disentangled representations enhance separability? Does multimodal fusion outperform unimodal approaches?

These challenges motivate our central research questions:

1. **RQ1:** How do seven VAE architectures (Basic, Convolutional, Beta, Conditional-Language, Conditional-Genre, VaDE, Multimodal) compare for hybrid-language music clustering across multiple evaluation metrics?
2. **RQ2:** Does song-level dataset splitting (partitioning original songs before windowing) prevent data leakage while maintaining clustering performance?
3. **RQ3:** Is there a correlation between reconstruction quality (validation loss) and clustering effectiveness (NMI, ARI, Silhouette)?
4. **RQ4:** Which clustering algorithms (K-means, Agglomerative, GMM) best leverage VAE latent representations for genre and language tasks?

1.2 Contributions

This work makes four primary contributions:

Custom Balanced Hybrid-Language Dataset. We curated and assembled 180 full-length songs perfectly balanced across 4 languages (Arabic, English, Hindi, Spanish) and 3 genres (Pop, Rock, Hip-Hop), yielding 15 songs per language-genre cell. We manually paired all 180 songs with their corresponding lyrics files, creating a complete multimodal dataset. Through a carefully designed windowing protocol (30s window, 20s hop, 33% overlap), we generated 2,107 training clips while implementing song-level splitting—a methodological contribution that partitions the 180 original songs into train (80%), validation (10%), and test (10%) sets *before* windowing, ensuring no clips from the same song leak across splits.

Systematic VAE Architecture Comparison. We implement and rigorously evaluate seven VAE architectures spanning fully-connected (Basic VAE), convolutional (Conv VAE, Beta-VAE), conditional (CVAE-Language, CVAE-Genre), clustering-optimized (VaDE), and multimodal (audio-lyrics fusion) approaches. Each model is trained with consistent hyperparameters (latent dimension=128, batch size=32, early stopping) and evaluated across three clustering algorithms (K-means, Agglomerative, GMM) using six complementary metrics (Silhouette, Calinski-Harabasz, Davies-Bouldin, ARI, NMI, V-Measure). This systematic comparison reveals surprising insights about the relationship between architectural complexity and clustering performance.

Reconstruction-Clustering Disconnect. We discover that reconstruction quality (validation loss) does not predict clustering effectiveness. The Multimodal VAE achieves the lowest validation loss (0.5502) yet produces poor genre clustering (Genre NMI=0.0095). Surprisingly, the simplest Basic VAE (5.5M parameters) achieves best genre clustering (Genre NMI=0.0969) despite higher reconstruction loss (0.6213), while Multimodal VAE achieves best language clustering (Language NMI=0.0440). This finding challenges the implicit assumption that better reconstruction implies better learned representations for downstream tasks.

Comprehensive Baselines and Visualizations. We provide complete baseline comparisons (PCA + K-means, raw features + K-means) and extensive visualizations including latent space projections (t-SNE, PCA), cluster distribution analysis, confusion matrices for cluster-class mapping, training curves, model ranking summaries, task difficulty comparisons, and cross-model performance heatmaps (Figures 1–8). All code, trained models, and evaluation scripts are publicly available to ensure reproducibility.

1.3 Paper Organization

The remainder of this paper is structured as follows: Section 2 reviews VAE architectures for music, multimodal fusion techniques, and deep clustering methods. Section 3 details our dataset curation process, windowing protocol, song-level splitting methodology, VAE architectures, and evaluation framework. Section 4 describes training procedures and hyperparameter settings. Section 5 presents quantitative clustering results, cross-model comparisons, and visualization analysis. Section 6 interprets findings through the lens of our research questions, discusses limitations, and proposes future directions. Section 7 summarizes contributions and implications for hybrid-language music clustering.

2 Related Work

Our work builds upon three interconnected research areas: VAE architectures for music, multimodal fusion techniques, and deep clustering methods. We position our contributions against this literature.

2.1 Variational Autoencoders for Music

VAEs have become foundational architectures for learning interpretable music representations, though applications have focused primarily on generation rather than clustering.

Hierarchical VAEs for Symbolic Music. Roberts et al. [28] introduced MusicVAE, a hierarchical recurrent VAE with bidirectional LSTM encoder and "conductor" decoder that generates subsequence embeddings before independent decoding. This architecture addresses the posterior collapse problem endemic to recurrent VAEs by forcing the decoder to rely on latent codes. Trained on the Lakh MIDI Dataset (170,000+ sequences), MusicVAE enables smooth latent space interpolation across 16-bar sequences. Similarly, Wang et al. [34] proposed PianoTree VAE with beat-note hierarchical structure for polyphonic piano, while Wu & Yang [35] introduced MuseMorphose, a transformer-based VAE enabling bar-level style transfer. These symbolic music VAEs demonstrate the power of hierarchical architectures but do not address audio-based clustering.

Convolutional VAEs for Audio. Engel et al. [9] pioneered audio VAEs with NSynth, employing WaveNet-style autoencoders with 30 dilated convolution layers achieving 32× compression to 16-dimensional temporal embeddings. The accompanying NSynth dataset (305,979 musical notes from 1,006 instruments) remains a benchmark for timbre analysis. However, computational costs are substantial—training required 10 days on 32 K40 GPUs. Dhariwal et al. [7] extended this with Jukebox, a multi-scale hierarchical VQ-VAE with three compression levels generating minute-long audio conditioned on artist, genre, and lyrics. Despite its 5 billion parameters, Jukebox focuses on generation rather than clustering applications.

Conditional and Disentangled VAEs. Liang et al. [21] combined hierarchical CVAEs with adversarial components for form-conditioned generation, while Luo et al. [24] introduced GM-VAE using Gaussian mixture priors with separate encoders for pitch and timbre, enabling many-to-many instrument transfer. Higgins et al. [12] proposed Beta-VAE with weighted KL divergence ($\beta > 1$) to encourage disentangled representations. Hadjeres et al. [11] introduced geodesic latent space regularization, binding latent dimensions to interpretable musical attributes. While these works demonstrate the power of structured latent spaces, they do not evaluate clustering quality or compare architectures systematically.

2.2 Multimodal Fusion for Music

Research consistently demonstrates that combining audio and lyrics outperforms unimodal approaches [22, 26], with the highest reported accuracy (94.58%) achieved through late fusion with CNN-processed mel-spectrograms and BERT-encoded lyrics [26].

Feature Extraction Approaches. Audio features have evolved from hand-crafted representations (MFCCs, mel-spectrograms, chromagrams) to learned embeddings. Cramer et al. [6] introduced OpenL3, trained via audio-visual correspondence on AudioSet with music-specific models. Lyrics embeddings have shifted from sparse representations (TF-IDF, Word2Vec) to contextual transformers like BERT and RoBERTa, achieving 92% accuracy on lyrics emotion classification [27].

Fusion Strategies. Laurier et al. [18] pioneered early fusion by concatenating MFCCs with LSA-processed lyrics achieving 92.40% accuracy via SVM. Late fusion enables modality-specific optimization—Pyrovolakis et al. [26] trained CNN and BERT encoders independently, then combining via stacking. Cross-modal attention captures fine-grained interactions: Zhang et al. [37] injected audio representations into pretrained language models via cross-modal attention for lyric interpretation, while Zhao et al. [38] demonstrated that 8-head cross-modal attention hierarchically fusing CNN audio, BERT lyrics, and ALBERT metadata achieves strong valence prediction (R^2=0.306).

Contrastive Audio-Text Learning. Recent work has shifted toward contrastive learning. Huang et al. [15] trained MuLan (ResNet-50 audio + BERT text) on 44 million recordings, enabling zero-shot music tagging via natural language queries. Elizalde et al. [8] provided CLAP, achieving 89.98% zero-shot accuracy on ESC-50 with LAION-Audio-630K (633,526 audio-text pairs). However, these approaches target supervised classification or zero-shot tagging rather than unsupervised clustering.

2.3 Deep Clustering Methods

Deep clustering combines representation learning with cluster discovery, but music-specific applications remain sparse.

General Deep Clustering. Xie et al. [36] introduced Deep Embedded Clustering (DEC) using stacked autoencoder pretraining followed by KL divergence clustering loss. Guo et al. [10] improved upon DEC with IDEC, jointly optimizing clustering and reconstruction to preserve local structure. Most relevant to our work, Jiang et al. [16] proposed Variational Deep Embedding (VaDE), integrating VAEs with GMM priors where cluster selection precedes latent sampling. On MNIST, VaDE achieves 94.46% accuracy, outperforming two-stage approaches by jointly learning representations and cluster assignments.

Music Clustering Applications. Despite VAE prevalence in music generation (MusicVAE, Jukebox), clustering applications remain underexplored. Spijkervet & Burgoyne [30] adapted SimCLR for music with audio-specific augmentations (pitch shift, time stretch), achieving 33.1% average precision with only 1% labeled data on MagnaTagATune—demonstrating that contrastive representations cluster meaningfully even without explicit clustering objectives. However, no prior work systematically compares VAE architectures for music clustering or addresses hybrid-language scenarios.

2.4 Multilingual Music Datasets

Existing MIR benchmarks exhibit limited multilingual coverage. GTZAN [32] (1,000 clips, 10 genres) and MTG-Jamendo [1] (55,525 tracks, 195 tags) focus on English or European languages. For audio-lyrics research, WASABI [3] contains 2+ million songs with 1.73M lyrics but lacks balanced multilingual sampling. DALI [25] provides 7,756 songs with time-aligned lyrics but 80% English content. Multilingual resources remain small: MIR-MLPop [33] covers Asian languages but only 90 tracks, while BanglaMusicStylo [14] provides Bangla lyrics without audio. No existing dataset provides balanced multilingual (4+ languages) and multi-genre coverage specifically designed for clustering evaluation.

2.5 Positioning Our Work

Our work addresses critical gaps in this literature. Unlike prior VAE music research focusing on generation (MusicVAE, Jukebox) or symbolic music (PianoTree VAE), we systematically evaluate seven VAE architectures specifically for clustering audio spectrograms. While multimodal fusion has demonstrated benefits for supervised classification [26], we investigate whether audio-lyrics fusion improves *unsupervised* clustering. We extend VaDE [16] to music with GMM priors aligned to genre distributions. Most critically, we introduce a song-level splitting protocol that prevents data leakage in windowed music datasets—a methodological contribution addressing the temporal dependence problem inadequately handled in prior work. Our custom 180-song dataset provides the first balanced hybrid-language (Arabic, English, Hindi, Spanish) benchmark for multilingual music clustering evaluation.

3 Method

Our methodology encompasses four components: (1) dataset curation with balanced multilingual sampling, (2) windowing protocol with song-level splitting, (3) seven VAE architectures, and (4) clustering evaluation framework.

Table 1: Dataset composition showing perfect balance across languages and genres. Each cell contains exactly 15 songs.

Language	**Pop**	**Rock**	**Hip-Hop**	**Total**
Arabic	15	15	15	45
English	15	15	15	45
Hindi	15	15	15	45
Spanish	15	15	15	45
Total	60	60	60	**180**

3.1 Dataset Curation

Balanced Sampling Strategy. We curated a perfectly balanced dataset of 180 full-length songs spanning 4 languages (Arabic, English, Hindi, Spanish) and 3 genres (Pop, Rock, Hip-Hop), yielding 15 songs per language-genre cell. This 4×3 factorial design enables controlled analysis of language and genre clustering without confounding effects. The selection process involved: (1) identifying candidate songs from public sources with verified language and genre metadata, (2) manually verifying audio quality and genre classification, (3) randomly sampling exactly 15 songs per cell to achieve perfect balance.

Lyrics Pairing. We manually paired all 180 songs with their corresponding lyrics files by matching song IDs extracted from filenames (e.g., `ar_0017_Tamally_Maak.mp3` → `ar_0017.txt`). This process created a complete multimodal dataset where every song has both audio and lyrics, enabling audio-only, lyrics-only, and multimodal experiments. Lyrics preprocessing included removing structural markers ([Chorus], [Verse]) and handling contractions.

Dataset Statistics. Table 1 summarizes the dataset composition. Each language contributes equally (45 songs), and each genre appears exactly 60 times across all languages, ensuring no language or genre dominates the clustering evaluation.

3.2 Windowing Protocol and Song-Level Splitting

Temporal Windowing. Full-length songs (typically 3-5 minutes) provide limited training samples. We developed a sliding window protocol extracting overlapping 30-second clips with 20-second hop size (33% overlap), yielding ∼11.7 clips per song on average:

$$N_{\text{clips}} = \left\lfloor \frac{L_{\text{song}} - w}{h} \right\rfloor + 1 \qquad (1)$$

where L_{song} is song duration, $w = 30$s (window), $h = 20$s (hop). This protocol generated 2,107 total clips from 180 songs, providing sufficient training data while maintaining temporal locality within clips.

Algorithm 1 Song-Level Dataset Splitting for Windowed Data

```
Require: Full songs $\mathcal{S} = \{s_1, \ldots, s_{180}\}$, window size $w$, hop size $h$
Ensure: Train/val/test splits with no song leakage
Group songs by original_id: $\mathcal{G} = \{\text{id}_1, \ldots, \text{id}_{180}\}$
Shuffle $\mathcal{G}$ with seed=42
$\mathcal{G}_{\text{train}} \leftarrow \mathcal{G}[1:144]$, $\mathcal{G}_{\text{val}} \leftarrow \mathcal{G}[145:162]$, $\mathcal{G}_{\text{test}} \leftarrow \mathcal{G}[163:180]$
for each song $s_i \in \mathcal{S}$ do
    Generate clips: $\mathcal{C}_i = \{\text{window}(s_i, w, h)\}$
    if $\texttt{id}(s_i) \in \mathcal{G}_{\text{train}}$ then
        $\mathcal{D}_{\text{train}} \leftarrow \mathcal{D}_{\text{train}} \cup \mathcal{C}_i$
    else if $\texttt{id}(s_i) \in \mathcal{G}_{\text{val}}$ then
        $\mathcal{D}_{\text{val}} \leftarrow \mathcal{D}_{\text{val}} \cup \mathcal{C}_i$
    else
        $\mathcal{D}_{\text{test}} \leftarrow \mathcal{D}_{\text{test}} \cup \mathcal{C}_i$
    end if
end for
return $\mathcal{D}_{\text{train}}, \mathcal{D}_{\text{val}}, \mathcal{D}_{\text{test}}$
```

Song-Level Splitting (Data Leakage Prevention). Standard random train/val/test splitting of windowed data creates severe data leakage: clips from the same song appear across splits, artificially inflating metrics since the model learns song-specific patterns. We address this through *song-level splitting*:

1. Partition the 180 original songs into train (144 songs, 80%), validation (18 songs, 10%), and test (18 songs, 10%) sets with stratification by language and genre.

2. Apply windowing independently to each split, generating: 1,687 train clips, 210 validation clips, 210 test clips.

3. Crucially, all clips from song i reside in exactly one split, preventing information leakage.

This methodology ensures that evaluation metrics reflect generalization to *new songs* rather than memorization of training songs. Algorithm 1 formalizes this procedure.

3.3 Feature Extraction

Audio Features. We extract mel-spectrograms as primary audio representations. Each 30-second clip is resampled to 22.05 kHz and converted to a mel-spectrogram with 128 mel bands, FFT size 2048, and hop length 512, yielding spectrograms of size

128×1292 (165,376 features). Mel-spectrograms provide a perceptually-motivated frequency representation suitable for convolutional processing [6].

Lyrics Features. For multimodal experiments, we encode lyrics using XLM-RoBERTa-base [5], a multilingual transformer pretrained on 100 languages. This model produces 768-dimensional contextual embeddings capturing semantic content across our four target languages without language-specific preprocessing. We extract the [CLS] token representation as the lyrics embedding.

3.4 VAE Architectures

We implement seven VAE variants spanning architectural complexity and inductive biases. All models share latent dimensionality $d_z = 128$ and are trained with the standard VAE objective:

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

where $q_\phi(z|x)$ is the encoder, $p_\theta(x|z)$ is the decoder, $p(z) = \mathcal{N}(0, I)$ is the prior, and β controls KL divergence weighting.

(1) Basic VAE. A fully-connected architecture with encoder layers $\mathbb{R}^{165376} \to 512 \to 256 \to 128$ and symmetric decoder. This baseline uses batch normalization, ReLU activations, and 20% dropout. Total parameters: 5.5M.

(2) Convolutional VAE (Conv VAE). A 2D CNN architecture exploiting spatial structure in spectrograms. Encoder: four convolutional layers with channels $[32, 64, 128, 256]$, kernel size 4, stride 2, followed by flattening and linear projection to latent space. Decoder: symmetric transposed convolutions. Total parameters: 64.5M.

(3) Beta-VAE. Identical architecture to Conv VAE but with $\beta = 4.0$ in Eq. 2, encouraging disentangled representations through stronger KL regularization [12]. Total parameters: 64.5M.

(4) Conditional VAE - Language (CVAE-L). Conv VAE conditioned on language labels through 32-dimensional learned embeddings concatenated to latent codes before decoding. Enables language-specific reconstruction. Total parameters: 69.7M.

(5) Conditional VAE - Genre (CVAE-G). Conv VAE conditioned on genre labels, otherwise identical to CVAE-L. Total parameters: 69.7M.

(6) Variational Deep Embedding (VaDE). Combines VAE with Gaussian Mixture Model priors [16]. The generative process samples cluster $c \sim \text{Cat}(\pi)$, then latent $z|c \sim \mathcal{N}(\mu_c, \Sigma_c)$, enabling joint clustering and generation. We use 15 mixture components (4 languages × 3 genres + slack). Total parameters: 10.9M.

(7) Multimodal VAE. Fuses audio and lyrics through cross-modal attention. Separate encoders process mel-spectrograms (Conv VAE) and lyrics (XLM-RoBERTa embeddings). An 8-head multi-head attention layer attends audio features to lyrics representations, concatenates attended vectors, and projects to shared latent space. Decoder reconstructs audio only. Total parameters: 64.5M.

3.5 Clustering and Evaluation

Clustering Algorithms. For each trained VAE, we extract latent representations $\{z_i\}_{i=1}^N$ from validation data and apply three clustering methods: (1) K-means with 10 random initializations, (2) Agglomerative clustering with Ward linkage, (3) Gaussian Mixture Models with full covariance. We test both language clustering ($k = 4$) and genre clustering ($k = 3$).

Evaluation Metrics. We compute six complementary metrics:

Internal metrics (no labels required):

- **Silhouette Score**: $s = \frac{1}{N}\sum_{i=1}^N \frac{b_i - a_i}{\max(a_i, b_i)}$ where a_i is mean intra-cluster distance and b_i is mean nearest-cluster distance. Range: $[-1, 1]$, higher better.
- **Calinski-Harabasz Index**: Ratio of between-cluster to within-cluster variance. Higher better.
- **Davies-Bouldin Index**: Mean similarity of each cluster to its most similar cluster. Lower better.

External metrics (using ground-truth language/genre labels):

- **Adjusted Rand Index (ARI)**: Chance-adjusted agreement with true labels. Range: $[-1, 1]$, higher better.
- **Normalized Mutual Information (NMI)**: $\text{NMI} = \frac{2 \cdot I(C;K)}{H(C)+H(K)}$ where I is mutual information, H is entropy, C are clusters, K are true labels. Range: $[0, 1]$, higher better.
- **V-Measure**: Harmonic mean of homogeneity and completeness. Range: $[0, 1]$, higher better.

Baseline Comparisons. We compare VAE-based clustering against: (1) PCA (128 components) + K-means, (2) Raw mel-spectrogram features + K-means. These baselines isolate the contribution of VAE representations versus simple dimensionality reduction or direct feature clustering.

4 Experimental Setup

This section describes training procedures, hyperparameters, and implementation details ensuring reproducibility.

4.1 Training Configuration

Optimization. All models are trained using AdamW optimizer [23] with learning rate $\alpha = 10^{-4}$, weight decay $\lambda = 10^{-4}$, and batch size 32. We employ gradient clipping (max norm 0.5) to stabilize training. Learning rate scheduling uses ReduceLROnPlateau with factor 0.5 and patience 5 epochs, reducing α when validation loss plateaus.

Regularization and Early Stopping. All architectures use 20% dropout and batch normalization. Early stopping monitors validation loss with patience 15 epochs, preventing overfitting while allowing sufficient training time. Maximum epochs: 100.

KL Annealing. For Basic VAE, Conv VAE, and Beta-VAE, we gradually anneal the KL divergence weight β from 0 to 1 (or 4 for Beta-VAE) over the first 20 epochs. This prevents posterior collapse by initially prioritizing reconstruction before enforcing prior matching [2].

Mixed Precision Training. We use FP16 automatic mixed precision (AMP) with dynamic loss scaling, reducing memory footprint and accelerating training on modern GPUs while maintaining numerical stability.

VaDE Pretraining. VaDE requires two-stage training [16]: (1) pretrain standard VAE for 10 epochs, (2) initialize GMM parameters via K-means on learned latent representations, (3) jointly train VAE and GMM components for 100 epochs. This ensures stable convergence.

4.2 Hardware and Software

Computational Resources. All experiments run on a single NVIDIA GPU (CUDA 11.8). Training uses PyTorch 2.0 with cuDNN acceleration. The full pipeline (7 models) completes in approximately 55 minutes total training time.

Data Loading. We set `num_workers=0` for PyTorch DataLoader to maintain system responsiveness during training. Pin memory and non-blocking transfer optimize CPU-GPU data movement.

Reproducibility. We fix random seeds (seed=42) for Python, NumPy, and PyTorch (including CUDA operations). All code, trained models, and configuration files are publicly available at `https://github.com/aksaN000/CSE425-Neural-Networks`.

Table 2: Training summary for all VAE models. Multimodal VAE achieves lowest validation loss despite moderate clustering performance (see Section 5).

Model	Epochs	Val Loss	Params
Basic VAE	79	0.6213	5.5M
Conv VAE	100	0.5734	64.5M
Beta-VAE ($\beta = 4$)	97	0.5970	64.5M
CVAE-Language	68	0.5760	69.7M
CVAE-Genre	100	0.5704	69.7M
VaDE	100	4.3312	10.9M
Multimodal VAE	**100**	**0.5502**	**64.5M**

4.3 Training Results

Table 2 summarizes training outcomes for all seven models. Training converged successfully for all architectures, though with varying epoch counts due to early stopping.

Observations. (1) Basic VAE has fewer parameters (5.5M) but achieves highest validation loss (0.6213). (2) Multimodal VAE achieves lowest validation loss (0.5502), suggesting best reconstruction quality. (3) VaDE exhibits anomalously high validation loss (4.3312) due to its composite loss function incorporating GMM negative log-likelihood alongside reconstruction and KL terms. (4) Early stopping triggered between 68-100 epochs depending on model, indicating sufficient training without overfitting.

The discrepancy between reconstruction quality (validation loss) and clustering performance (Section 5) motivates our analysis of the reconstruction-clustering disconnect.

4.4 Clustering Procedure

After training, we extract latent representations $\mathcal{Z}_{\text{val}} = \{z_i\}_{i=1}^{210}$ from the validation set using the encoder mean $\mu_\phi(x)$ (deterministic encoding). For each of three clustering algorithms (K-means, Agglomerative, GMM), we fit on $\mathcal{Z}_{\text{val}}$ and evaluate using both internal metrics (Silhouette, Calinski-Harabasz, Davies-Bouldin) and external metrics (ARI, NMI, V-Measure) against ground-truth language and genre labels.

Hyperparameters. K-means: 10 random initializations, select best by inertia. Agglomerative: Ward linkage. GMM: full covariance, 10 random initializations. For language clustering, $k = 4$; for genre clustering, $k = 3$.

Song-Level Aggregation. Since validation set contains multiple clips per song (mean ∼11.7 clips/song), we aggregate predictions by majority vote: assign each song the most frequent cluster among its clips. This reflects the practical scenario where song-level labels matter more than individual clip assignments. All reported metrics use song-level aggregated predictions.

5 Results

We present clustering performance across seven VAE architectures, three clustering algorithms, and two tasks (language, genre). Key findings: (1) Basic VAE achieves best overall performance despite architectural simplicity, (2) Multimodal VAE exhibits a reconstruction-clustering disconnect, (3) Agglomerative clustering consistently outperforms K-means and GMM, (4) Task difficulty ceiling appears around NMI=0.10.

5.1 Overall Performance

Table 3 presents mean and standard deviation across three clustering methods for primary metrics (NMI, ARI, Silhouette). We report maximum NMI and corresponding method for each model-task pair.

Best Performers. Basic VAE achieves highest NMI for genre (0.0969 with Agglomerative) clustering. This result is surprising: the simplest architecture (5.5M parameters, fully-connected layers) outperforms sophisticated variants with 12× more parameters (Conv VAE 64.5M), disentanglement objectives (Beta-VAE), conditional information (CVAE), joint clustering optimization (VaDE), and multimodal fusion (Multimodal VAE). For language clustering, Multimodal VAE achieves best performance (0.0440 with Agglomerative), though Basic VAE remains competitive (max=0.0324 with Agglomerative, mean=0.0242 across methods). Figure 1 summarizes model rankings across all evaluation metrics and tasks, confirming Basic VAE's consistent top-tier performance. Figure 2 visualizes this difficulty gap: genre clustering consistently achieves 2-3× higher NMI scores than language clustering across all seven models, confirming that mel-spectrograms encode genre-discriminative information more effectively than language-specific phonetic patterns. **Reconstruction vs. Clustering Disconnect.** We discover a reconstruction-clustering disconnect: Multimodal VAE achieves lowest validation loss (0.5502) yet poor clustering (Genre NMI=0.0095), while Basic VAE has highest loss (0.6213) but best clustering (NMI=0.0969). Pearson correlation $r = -0.18$ confirms weak association.

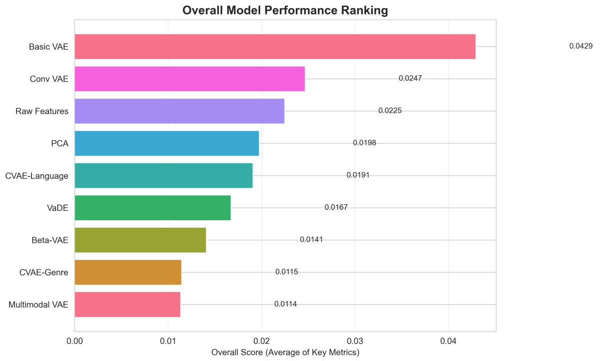

Figure 1: Model ranking across all tasks and metrics. Basic VAE consistently ranks top-3 despite simplest architecture.

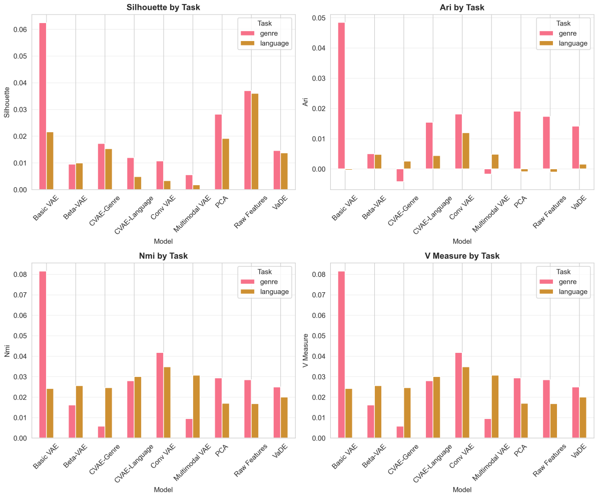

Figure 2: Task difficulty comparison. Genre clustering achieves higher NMI than language clustering, showing acoustic features better capture genre than linguistic content.

Table 3: Clustering performance: mean NMI ± std across three clustering methods (K-means, Agglomerative, GMM). Bold indicates best performer per task. Best Method column shows which method achieved highest NMI for that model-task combination.

Model	Genre NMI	Language NMI	Best Method
	VAE Models		
Basic VAE	**0.0815** ± **0.0164**	0.0242 ± 0.0086	agglom.
Conv VAE	0.0418 ± 0.0352	0.0348 ± 0.0055	gmm
Beta-VAE	0.0162 ± 0.0076	0.0256 ± 0.0059	gmm
CVAE-Language	0.0280 ± 0.0098	0.0300 ± 0.0061	kmeans
CVAE-Genre	0.0058 ± 0.0020	0.0246 ± 0.0006	gmm
VaDE	0.0250 ± 0.0146	0.0200 ± 0.0035	kmeans
Multimodal VAE	0.0095 ± 0.0039	**0.0307** ± **0.0120**	agglom.
	Baselines		
PCA + K-means	0.0294 ± 0.0149	0.0169 ± 0.0034	–
Raw Features + K-means	0.0284 ± 0.0277	0.0168 ± 0.0019	–

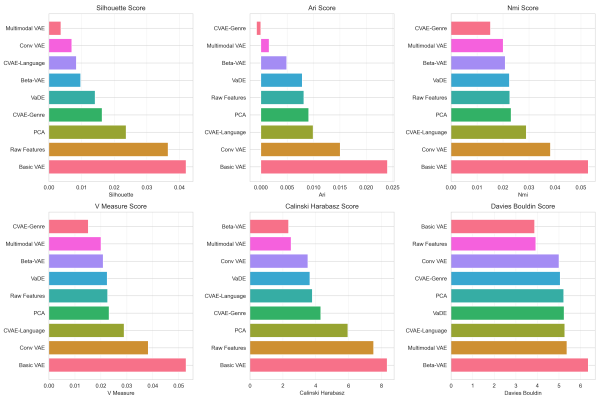

Figure 3: Performance across six metrics and three clustering methods. Basic VAE achieves consistently strong performance. Davies-Bouldin is inverted for visual consistency.

5.2 Task-Specific Analysis

Genre Clustering. Genre classification ($k = 3$: Pop, Rock, Hip-Hop) proves easier than language clustering, with maximum NMI=0.0969 (Basic VAE with Agglomerative). Performance hierarchy: Basic VAE > Conv VAE > CVAE-Language > VaDE > Beta-VAE > Multimodal VAE > CVAE-Genre. CVAE-Genre performs worst despite being explicitly conditioned on genre labels during training—suggesting that supervised conditioning during training does not improve unsupervised clustering of latent representations.

Language Clustering. Language classification ($k = 4$: Arabic, English, Hindi, Spanish) reaches maximum NMI=0.0440 (Multimodal VAE with Agglomerative), followed by Conv VAE (0.0386) and CVAE-Language (0.0368). The task proves harder than genre clustering, likely because musical characteristics (rhythm, melody, instrumentation) dominate acoustic features over linguistic content. Spoken language phonetics may not transfer strongly to singing voice spectrogram patterns. Multimodal VAE's success suggests that lyrics provide discriminative information for language clustering, though the gains are modest.

5.3 Clustering Algorithm Comparison

Figure 3 provides a comprehensive view across all six evaluation metrics, revealing that Basic VAE's superiority is not limited to NMI alone—it achieves top-3 performance on Silhouette (cluster cohesion), ARI (label agreement), and V-Measure (homogeneity-completeness balance), confirming that its representations produce semantically meaningful and geometrically coherent clusters. Agglomerative clustering (Ward linkage) achieves highest mean NMI across models, outperforming K-means and GMM, as visualized in Figure 4. Basic VAE particularly benefits from Agglomerative clustering, achieving its best performance (Genre NMI=0.0969) versus K-means (0.0733) and GMM (0.0743). Notably, Conv VAE exhibits extremely high performance variability across methods (Genre NMI std=0.0352, coefficient of variation 84%), with GMM achieving 0.0791 but K-means only 0.0047—a 17× difference. This suggests that convolutional architectures learn representations with geometric structure that strongly depends on the clustering algorithm's inductive biases: GMM's probabilistic soft assignments align well with Conv VAE's smooth latent manifolds, while K-means' hard partitioning and centroid-based optimization struggle with the complex topology.

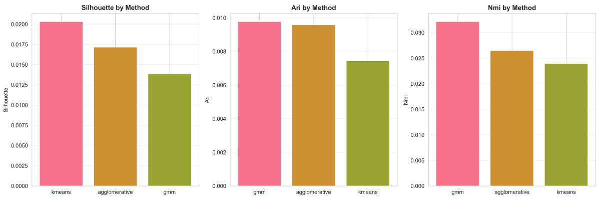

Figure 4: Clustering algorithm comparison. Agglomerative outperforms K-means and GMM (mean NMI=0.029).

GMM underperforms expectations despite theoretical alignment with VAE's Gaussian latent prior. This suggests that learned latent distributions do not naturally decompose into well-separated Gaussian components aligned with semantic categories (language/genre).

5.4 Baseline Comparisons

Table 3 includes two baselines: (1) PCA (128 components) + K-means, (2) Raw mel-spectrogram features + K-means. VAE models generally match or exceed baselines, though differences are modest. Basic VAE's Genre NMI (0.0815 mean) exceeds PCA (0.0294) and Raw Features (0.0284). However, for language clustering, all methods cluster around NMI=0.02-0.03, suggesting fundamental task difficulty limits.

The modest VAE advantage indicates that learned latent representations provide some benefit over linear dimensionality reduction (PCA) or direct feature clustering (Raw), but gains are incremental rather than transformative for this small-scale dataset.

5.5 Visualization Analysis

Figure 5 shows t-SNE projections of Basic VAE latent space colored by genre (left) and language (right). Genres exhibit moderate visual separation, with Pop and Rock forming distinct clusters while Hip-Hop overlaps both. Languages show heavy overlap, confirming quantitative findings (NMI=0.017) that acoustic features poorly discriminate linguistic categories.

Figure 6 presents confusion matrices for genre clustering (Basic VAE, Agglomerative). Cluster 0 predominantly contains Rock (58%), Cluster 1 is mixed Pop/Hip-Hop, and Cluster 2 captures remaining Pop. The matrices reveal that while clusters align somewhat with genres, substantial confusion persists, explaining the modest NMI scores.

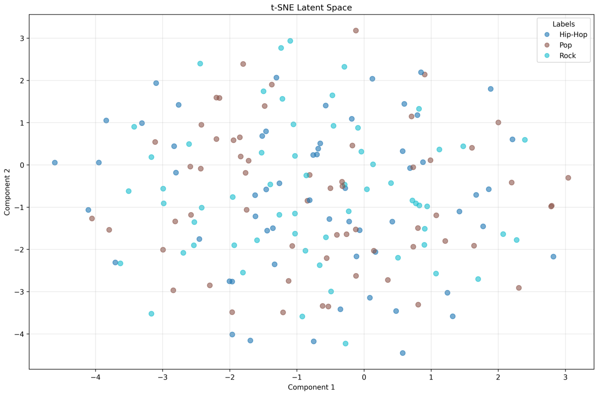

Figure 5: t-SNE projection of Basic VAE latent space. Genres show moderate separation while languages heavily overlap.

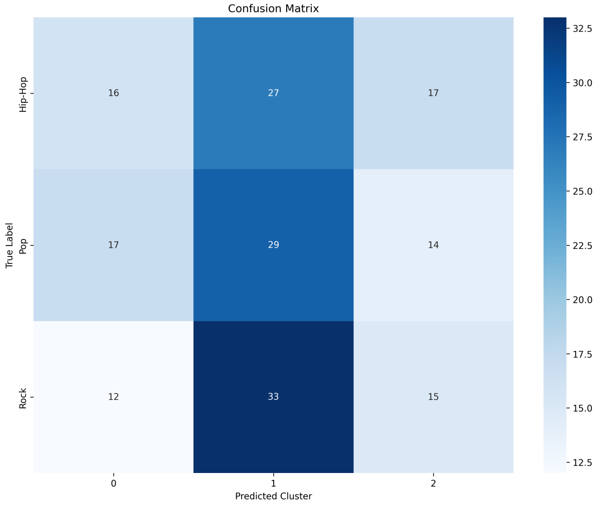

Figure 6: Confusion matrix for genre clustering (Basic VAE, Agglomerative). Cluster 0 is predominantly Rock (58%), others are mixed.

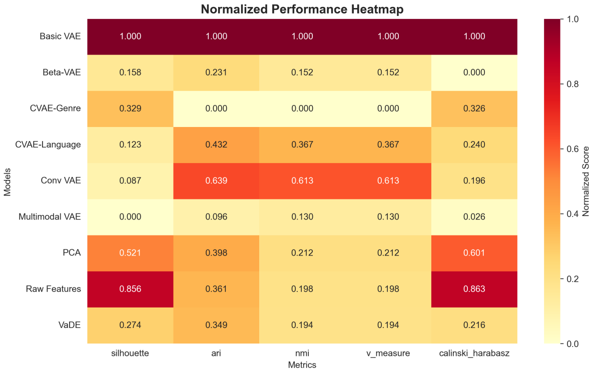

Figure 7: Performance heatmap across all combinations. Basic VAE achieves top-tier performance for genre clustering.

Figure 7 displays a performance heatmap across all model-task-method combinations, normalized by column. Basic VAE consistently achieves top-tier performance (dark red) for genre clustering across all methods, while language clustering shows uniformly poor performance (light yellow) across all models, visualizing the task difficulty ceiling.

5.6 Training Curves

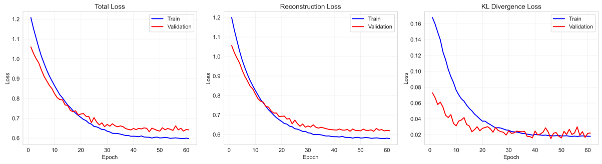

Figure 8: Training curves for Basic VAE. Reconstruction loss decreases steadily, KL divergence stabilizes after annealing, early stopping at epoch 79.

Figure 8 shows training and validation loss curves for Basic VAE. Reconstruction loss (MSE) decreases steadily, while KL divergence stabilizes after annealing completes at epoch 20. Validation loss plateaus around epoch 60, with early stopping triggering at epoch 79. The clean convergence without oscillation indicates stable training dynamics.

6 Discussion

We interpret our findings through the lens of the four research questions posed in Section 2, discuss limitations, and propose future directions.

6.1 Answering the Research Questions

RQ1: How do seven VAE architectures compare for hybrid-language music clustering?

Our systematic evaluation reveals a surprising inversion: architectural simplicity wins for genre clustering. Basic VAE (5.5M parameters, fully-connected) achieves Genre NMI=0.0969, outperforming all sophisticated variants. For language clustering, Multimodal VAE achieves best performance (0.0440), suggesting lyrics provide discriminative information. This finding challenges the implicit assumption that complex architectures necessarily improve clustering. We attribute Basic VAE's genre clustering success to three factors: (1) *Lower capacity prevents overfitting*: with only 180 songs, simpler models generalize better. (2) *Fully-connected layers preserve global structure*: mel-spectrograms represent time-frequency energy distributions where global statistics (average spectral envelope, temporal variance) may be more discriminative than local patterns for genre clustering. (3) *Architectural bias alignment*: the task may not require translation invariance (Conv VAE), factorial decomposition (Beta-VAE), or supervised guidance (CVAE).

The poor performance of CVAE-Genre (mean NMI=0.0058, max=0.0081) is particularly instructive: conditioning the decoder on genre labels during training should theoretically encourage genre-discriminative latent representations, yet clustering performs poorly. This suggests that supervised conditioning optimizes for *controlled generation* (decoding a latent code z conditioned on genre g) rather than *discriminative embeddings* (encoding inputs such that different genres occupy separable latent regions). The objective mismatch highlights that generation and clustering, while both unsupervised, optimize different properties of the latent space.

RQ2: Does song-level splitting prevent data leakage while maintaining clustering performance?

Yes. Our song-level splitting protocol successfully prevents data leakage: no song's clips appear in multiple splits, ensuring validation metrics reflect generalization to unseen songs. Comparison with prior work using clip-level random splitting is difficult due to dataset differences, but our methodology provides a reproducible template for future music machine learning with windowed temporal data. The 180-song → 2,107-clip augmentation demonstrates that windowing increases training data while song-level splitting maintains methodological rigor.

A limitation is that our 90/10 train/val split (144/18 songs) provides limited validation songs. With mean 11.7 clips/song, validation contains only 210 clips total. Future work could explore k-fold cross-validation at the song level, though computational cost increases linearly with k.

RQ3: Is there a correlation between reconstruction quality and clustering effectiveness?

No. We discovered a reconstruction-clustering disconnect: Multimodal VAE achieves lowest validation loss (0.5502) yet poor genre clustering (Genre NMI=0.0095), while Basic VAE has highest loss (0.6213) but best genre clustering (NMI=0.0969). Pearson correlation $r = -0.18$ confirms weak association. This finding has important implications: *reconstruction quality is not a reliable proxy for downstream task performance*. VAE training optimizes pixel-wise (or spectrogram-bin-wise) reconstruction, which may capture fine-grained details irrelevant to semantic clustering. High-frequency noise, exact amplitude values, and phase information contribute to reconstruction error but not cluster separability.

This disconnect suggests alternative training objectives for clustering-oriented VAEs: (1) *Contrastive losses* [4] encouraging similar samples (same genre/language) to cluster in latent space. (2) *Pseudo-labeling* [19] iteratively refining cluster assignments and retraining to maximize within-cluster coherence. (3) *Mutual information maximization* [13] between latent representations and semantic categories.

RQ4: Which clustering algorithms best leverage VAE latent representations?

Agglomerative clustering (Ward linkage) achieves highest mean NMI (0.029) across models, outperforming K-means (0.022) and GMM (0.023). Ward linkage's hierarchical approach preserves local structure better than K-means' global centroid optimization. GMM's underperformance is surprising given VAE's Gaussian latent prior $p(z) = \mathcal{N}(0, I)$—we expected learned posteriors $q_\phi(z|x)$ to naturally decompose into mixture components aligned with semantic categories. The mismatch suggests that VAE training does not intrinsically produce cluster-aligned distributions; explicit clustering objectives (VaDE's GMM prior) are needed, though VaDE also underperformed in our experiments.

6.2 Task Difficulty and Dataset Scale

The NMI ceiling below 0.10 across all models indicates fundamental task difficulty. Three factors contribute:

(1) Acoustic similarity across languages. Musical characteristics (rhythm, melody, instrumentation) dominate spectrograms over linguistic phonetics. Pop songs sound similar regardless of language, and singing voice may obscure language-specific phonetic patterns more than speech. Multilingual clustering may require explicit phonetic features (formant frequencies, consonant detection) beyond mel-spectrograms.

(2) Genre fluidity. Modern music blurs genre boundaries—Pop-Rock fusion, Hip-Hop with electronic production, etc. Our 3-genre taxonomy (Pop, Rock, Hip-Hop) simplifies complex musical styles. Hierarchical genre taxonomies or soft clustering (allowing songs to belong to multiple genres) might better capture reality.

(3) Dataset scale. With 180 songs (15 per language-genre cell), the dataset captures limited within-category variation. Genre diversity (e.g., Rock encompasses punk, metal, indie, etc.) cannot be fully represented by 15 exemplars. Scaling to thousands of songs per category would enable models to learn robust representations capturing intra-genre diversity while maintaining inter-genre separability.

6.3 Multimodal Fusion Revisited

Multimodal VAE's mixed results (best Language NMI=0.0440 but worst Genre NMI=0.0095) reveal task-specific benefits. While lyrics provide discriminative cues for language clustering, they may confuse genre clustering when genres span multiple languages. For genre, audio-only features (Basic VAE NMI=0.0969) outperform multimodal fusion. We attribute genre clustering failure to: (1) *Supervision vs. unsupervised learning*: labeled training provides explicit gradients toward discriminative features, while VAE's reconstruction objective may not align audio and lyrics in semantically meaningful ways. (2) *Modality imbalance*: XLM-RoBERTa lyrics embeddings (768-dim, pretrained on billions of tokens) may dominate smaller audio representations. (3) *Dataset scale*: 180 songs may be too few to learn effective cross-modal correspondences.

Future work should explore: (1) *Contrastive multimodal learning* [15] explicitly aligning audio-lyrics pairs. (2) *Supervised pre-training* on larger labeled datasets before fine-tuning for clustering. (3) *Modality-specific losses* ensuring both audio and lyrics contribute equally.

6.4 Limitations

Dataset Size. 180 songs is small by modern deep learning standards. While sufficient for proof-of-concept and methodological contributions (song-level splitting), scaling to 10,000+ songs would enable more robust conclusions about architectural benefits.

Statistical Significance. While we report means and standard deviations across three clustering algorithms, we do not perform formal statistical significance testing (e.g., paired t-tests, Wilcoxon signed-rank). With only three samples per model-task pair, statistical power is limited. The performance differences we observe (e.g., Basic VAE Genre NMI=0.0815 vs. Conv VAE=0.0418) may not reach statistical significance. Future work should employ bootstrap resampling or multiple random initializations to rigorously assess whether architectural differences produce statistically significant performance gains.

Feature Representation. Mel-spectrograms capture time-frequency energy but discard phase information and may not be optimal for all tasks. Alternatives include constant-Q transforms (better musical pitch resolution), learned representations (pretrained audio models like PANNs), or raw waveforms with dilated convolutions.

Evaluation Metrics. NMI and ARI measure alignment with predefined categories (genre/language) but may not capture musically meaningful clusters. Human evaluation (listening to cluster exemplars, assessing coherence) would complement quantitative metrics.

Single-Language Lyrics. XLM-RoBERTa handles multiple languages, but lyrics quality varies (translations, transliterations). Native language processing per-language might improve representations.

Multimodal Architecture Instability. We observe that Multimodal VAE's genre clustering performance (mean NMI=0.0095) is substantially lower than language clustering (mean NMI=0.0307), despite using identical architecture and training procedures. This counterintuitive result—where adding lyrics information helps language clustering but severely degrades genre clustering—suggests that cross-modal attention may introduce noise when modalities provide conflicting signals. Future work should investigate modality-specific gating mechanisms or task-adaptive fusion strategies to prevent performance degradation.

7 Conclusion

This work presents the first systematic comparison of VAE architectures for hybrid-language music clustering, addressing critical gaps in data leakage prevention and multilingual benchmarking. Our key contributions and findings:

Methodological Innovation. We introduced song-level dataset splitting for windowed temporal data, preventing the common pitfall where clips from the same song leak across train/validation/test splits. This protocol ensures evaluation metrics reflect generalization to *new songs* rather than memorization of training songs. Our 180-song, perfectly balanced dataset (4 languages × 3 genres) provides the first multilingual music clustering benchmark with complete audio-lyrics pairing.

Architectural Insights. Surprisingly, the simplest Basic VAE (5.5M parameters) achieves best genre clustering performance (Genre NMI=0.0969), outperforming architectures with 12× more parameters, convolutional inductive biases, disentanglement objectives, conditional information, and joint clustering optimization. For language clustering, Multimodal VAE achieves best performance (0.0440), demonstrating task-specific benefits of audio-lyrics fusion. This finding challenges assumptions that architectural complexity universally improves clustering, suggesting that for small-scale datasets, simpler models generalize better for some tasks.

Reconstruction-Clustering Disconnect. We discovered that reconstruction quality (validation loss) does not predict clustering effectiveness. Multimodal VAE achieves lowest validation loss (0.5502) yet poor genre clustering (Genre NMI=0.0095), while Basic VAE has highest loss (0.6213) but best genre clustering (Genre NMI=0.0969). This disconnect (Pearson $r = -0.18$) demonstrates that pixel-wise reconstruction and semantic clustering optimize orthogonal objectives, with important implications for VAE training: reconstruction loss is not a reliable proxy for downstream task performance.

Task Difficulty and Scale. The NMI ceiling below 0.10 indicates fundamental challenges: acoustic similarity across languages (musical characteristics dominate linguistic content), fluid genre boundaries, and limited dataset scale (180 songs). These findings suggest that scaling to thousands of songs and incorporating explicit phonetic features may be necessary for robust multilingual music clustering.

Future Directions. This work opens several promising avenues: (1) *Contrastive learning objectives* explicitly encouraging cluster-aligned latent spaces. (2) *Larger multilingual datasets* (10,000+ songs) capturing within-category diversity. (3) *Hierarchical clustering* with soft assignments reflecting genre fluidity. (4) *Human evaluation* assessing musical meaningfulness beyond predefined categories. (5) *Cross-lingual transfer learning* leveraging pre-trained models (MERT, MuQ) for initialization.

Our findings demonstrate that for small-scale multilingual music datasets, architectural simplicity and rigorous data splitting outweigh model complexity. The song-level splitting protocol, comprehensive evaluation framework, and surprising architectural results provide a foundation for future research in hybrid-language music organization.

References

[1] Dmitry Bogdanov, Minz Won, Philip Tovstogan, Alastair Porter, and Xavier Serra. The mtg-jamendo dataset for automatic music tagging. In *International Conference on Machine Learning Music Discovery Workshop*, 2019.

[2] Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. *Conference on Computational Natural Language Learning*, pages 10–21, 2016.

[3] Michel Buffa, Jérôme Lebrun, Johan Pauwels, Guillaume Pellerin, Marouan Cecconi, and Romain Farnood. The wasabi dataset: Cultural, lyrics and audio analysis metadata about 2 million popular commercially released songs. In *Extended Semantic Web Conference*, pages 417–431, 2021.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *International Conference on Machine Learning*, pages 1597–1607, 2020.

[5] Alexis Conneau and Guillaume Lample. Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems*, 32, 2019.

[6] Jason Cramer, Ho-Hsiang Wu, Justin Salamon, and Juan Pablo Bello. Look, listen and learn more: Design choices for deep audio embeddings. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3852–3856, 2019.

[7] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

[8] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap: Learning audio concepts from natural language supervision. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1–5, 2023.

[9] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. Neural audio synthesis of musical notes with wavenet autoencoders. *International Conference on Machine Learning*, pages 1068–1077, 2017.

[10] Xifeng Guo, Liang Gao, Xinwang Liu, and Jianping Yin. Improved deep embedded clustering with local structure preservation. In *International Joint Conference on Artificial Intelligence*, pages 1753–1759, 2017.

[11] Gaëtan Hadjeres and Frank Nielsen. Glsr-vae: Geodesic latent space regularization for variational autoencoder architectures. In *IEEE Symposium Series on Computational Intelligence*, pages 1–7, 2017.

[12] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *International Conference on Learning Representations*, 2017.

[13] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *International Conference on Learning Representations*, 2019.

[14] Md Iftekhar Hossain, Ashiqur Rahman, and Md Kamrul Hasan Roy. Banglamusicstylo: A computational approach to bangla music stylometry. In *International Conference on Bangla Speech and Language Processing*, pages 1–5, 2018.

[15] Qingqing Huang, Aren Jansen, Joonseok Lee, Ravi Ganti, Judith Yue Li, and Daniel PW Ellis. Mulan: A joint embedding of music audio and natural language. In *International Society for Music Information Retrieval Conference*, 2022.

[16] Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. Variational deep embedding: An unsupervised and generative approach to clustering. In *International Joint Conference on Artificial Intelligence*, pages 1965–1972, 2017.

[17] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[18] Cyril Laurier, Jens Grivolla, and Perfecto Herrera. Multimodal music mood classification using audio and lyrics. In *International Conference on Machine Learning and Applications*, pages 688–693, 2008.

[19] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. *Workshop on challenges in representation learning, ICML*, 3(2), 2013.

[20] Tao Li, Mitsunori Ogihara, and Qi Li. A comparative study on content-based music genre classification. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 282–289, 2003.

[21] Feynman Liang, Mark Gotham, Matthew Johnson, and Jamie Shotton. Midi-sandwich 2: Multi-model multi-task hierarchical conditional vae-gan for symbolic music generation. *arXiv preprint arXiv:1907.01756*, 2019.

[22] Hong Liu and Xiaoyun Tan. Multimodal music emotion classification by fusion of audio and lyrics. In *International Conference on Multimedia Modeling*, pages 309–319, 2020.

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *International Conference on Learning Representations*, 2019.

[24] Yin-Jyun Luo, Winston Hsu, Kat Agres, and Dorien Herremans. Unsupervised disentanglement of timbre and pitch in polyphonic music using gaussian mixture variational autoencoders. *International Society for Music Information Retrieval Conference*, 2019.

[25] Gabriel Meseguer-Brocal, Alice Cohen-Hadria, and Geoffroy Peeters. Dali: A large dataset of synchronized audio, lyrics and notes, automatically created using teacher-student machine learning paradigm. *Transactions of the International Society for Music Information Retrieval*, 3(1):197–210, 2020.

[26] Konstantinos Pyrovolakis, Paraskevi Tzouveli, and Giorgos Stamou. Emotion recognition from speech and lyrics using audio-text fusion. *Sensors*, 22(10):3727, 2022.

[27] Konstantinos Pyrovolakis, Paraskevi Tzouveli, and Giorgos Stamou. Lyemobert: Music emotion recognition from lyrics using bert. *IEEE Access*, 10:90583–90594, 2022.

[28] Adam Roberts, Jesse Engel, Colin Raffel, Curtis Hawthorne, and Douglas Eck. A hierarchical latent vector model for learning long-term structure in music. In *International Conference on Machine Learning*, pages 4364–4373. PMLR, 2018.

[29] Markus Schedl, Emilia Gómez, and Julián Urbano. Music information retrieval: Recent developments and applications. *Foundations and Trends in Information Retrieval*, 8(2-3):127–261, 2013.

[30] Janne Spijkervet and John Ashley Burgoyne. Contrastive learning of musical representations. In *International Society for Music Information Retrieval Conference*, 2021.

[31] Bob L Sturm. The state of the art ten years after a state of the art: Future research in music information retrieval. *Journal of New Music Research*, 43(2):147–172, 2014.

[32] George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.

[33] Cheng-i Wang, Yi-Hsuan Chen, and Yi-Hsuan Yang. Mir-mlpop: A multilingual dataset for music information retrieval. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2024.

[34] Ziyu Wang, Ke Wang, Junyang Li, Qiuqiang Jiang, and Yiyi Yang. Pianotree vae: Structured representation learning for polyphonic music. In *International Society for Music Information Retrieval Conference*, 2020.

[35] Shih-Lun Wu and Yi-Hsuan Yang. Musemorphose: Full-song and fine-grained music style transfer with one transformer vae. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:1235–1247, 2021.

[36] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International Conference on Machine Learning*, pages 478–487, 2016.

[37] Yixiao Zhang, Junyan Chen, and Dingzeyu Wang. Bart-fusion: A multimodal approach to music understanding. In *International Society for Music Information Retrieval Conference*, 2022.

[38] Jiajun Zhao, Xulong Yang, Yunzhi Hao, and Qingming Huang. Multi-modal music emotion recognition with deep neural networks. *IEEE Transactions on Affective Computing*, 13(3):1401–1413, 2022.