Integrating Symbolic and Waveform Music into Large Language Models

Teng Tu[1], Xiaohao Liu[1], Yunshan Ma[2], Ji Qi[3], and Tat-Seng Chua[1]

[1] National University of Singapore, Singapore
{teng.tu,xiaohao.liu}@u.nus.edu, dcscts@nus.edu.sg
[2] Singapore Management University, Singapore
ysma@smu.edu.sg
[3] Tsinghua University, Beijing, China
qj20@mails.tsinghua.edu.cn

Abstract. Music, as a unique and integral element of human life, is characterized by its complex structures, intricate details, and the fusion of multimodal information. Recent study advance music understanding by leveraging knowledge and reasoning capabilities derived from Large Language Models (LLMs). However, they often lack compatibility and fail to fully utilize the complementary strengths of diverse representations (e.g., ABC, MIDI, Waveform). To address these limitations, we propose a unified music-language model framework, named UniMuLM, transitioning from single-representation approaches to the integration of multiple music representations for LLM. Unifying different music representation formats poses challenges such as patch integrity and boundary ambiguity that arise from temporal discrepancies across these representations. To address these issues, UniMuLM employs a unified encoder that hierarchically aligns representations across multiple granularities, using contrastive learning and cross-reconstruction training to support coherent integration. Fine-tuned in multiple stages on open-source datasets, UniMuLM demonstrates the potential to handle dual-representation inputs. Notably, it achieves performance competitive with specialized waveform-only models on music understanding tasks, while surpassing open-source baselines in downstream applications such as music knowledge answering and ABC melody completion.

Keywords: Music Language Model · Music Understanding · Multimodal Language Model · Sound and Music Computing

1 Introduction

LLMs have achieved significant progress in linguistic tasks and also demonstrated potential in understanding other modalities (*e.g.*, vision), motivating the exploration of Multimodal Large Language Models [22]. Among various modalities, music stands out as a particularly challenging and underexplored domain due to

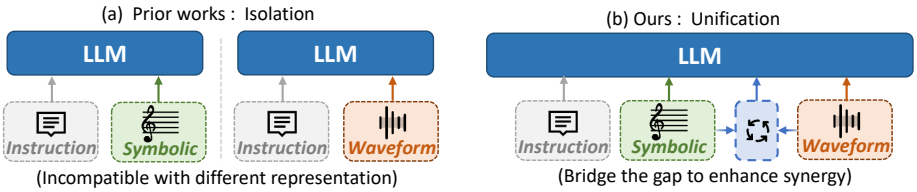

Fig. 1: Paradigm comparison between our method and prior works in music representation integration. (a) Prior works focus on utilizing either symbolic or waveform representations, typically in isolation. (b) In contrast, our approach introduces a unification mechanism bridging the gap and enhances their synergy.

its unique interplay of rhythm, melody, harmony, and lyrics, which collectively evoke human emotions. This has sparked significant research interest in integrating musical knowledge into LLMs for music understanding tasks, *e.g.*, music captioning, question answering, and reasoning [5,23,10]. In this paper, we refer to such models as Music Understanding Language Models (MuLMs), following the terminology introduced by MU-LLaMA [23].

A pivotal obstacle preventing MuLMs from approaching human-level expertise in music lies in the treatment of music representations, which are typically either symbolic notations [35,4] or performance waveforms [13,11], with most models designed to handle only one of these forms. These two representations are inherently complementary: symbolic music captures structural and harmonic relationships, while waveforms preserve timbral characteristics and performance nuances. This synergy mirrors how human musicians perceive music and underscores the potential for more comprehensive music modeling.

Despite the salient research gap, unifying both representation formats is nontrivial due to the temporal scale inconsistency between the two formats. Specifically, symbolic music is organized into uneven time segments based on note durations and maintains semi-structured boundaries between notes, bars, phrases and movements. In contrast, waveform signals are sampled at much higher frequencies with overlapping note sounds, making clear segmentation boundaries difficult to establish. While various efforts have been made to encode waveforms [28,34,7,9,21], none have effectively bridged the representational gap to enable consistent unified modeling. This limitation has prevented LLMs from learning both representation formats simultaneously, motivating our development of a unified music encoder.

Achieving this unification requires addressing two key technical challenges. The first challenge lies in the accurate integration of information during unification: conventional waveform patching methods rely on temporal segmentation [8,34], which complicates maintaining patch integrity while managing boundary ambiguities. Second, training stability when adapting language models for multiple modalities, particularly given the scarcity of training scenarios where symbolic music, waveform audio, and textual instructions co-occur.

To this end, we propose UniMuLM, a novel Music Understanding Language Model framework that employs a unified encoder compatible with both symbolic and waveform music representations. First, we train the encoder to seamlessly pre-align music representations, ensuring synchronization between different representation formats while preserving token efficiency. Specifically, we utilize a hierarchical aligning training mechanism that enhances representational capabilities at both high-level temporal scales and fine-grained levels. Subsequently, to adapt LLMs for music tasks, we begin by applying LoRA tuning [17] to leveraging music knowledge and symbolic music datasets [33,31] to infuse music knowledge into the LLM. Afterward, we adapt the unified encoder to handle diverse music representations in shared embedding space across all downstream tasks using various datasets [2,26,27,23,31]. Our contributions are threefold:

- We emphasize the often-overlooked complementarity of music representations, proposing UniMuLM as the first framework to integrate symbolic and waveform music.
- We introduce and train a unified encoder that hierarchically aligns different music representations, merging the advantages of efficiency and effectiveness across different granularities.
- We benchmark UniMuLM using four music understanding datasets and explore its capability in various downstream tasks.

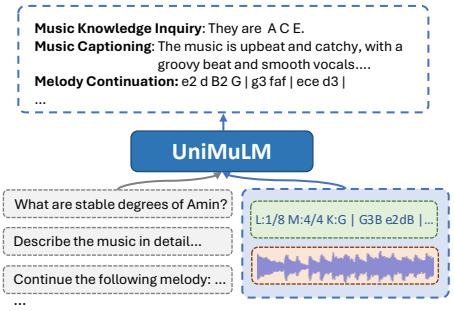

Fig. 2: A demonstration of UniMuLM in handling various types of music tasks.

2 Related Work

2.1 Music Encoding

Music representations in deep learning are broadly categorized into waveform and symbolic formats.

Waveform Representation consists of one-dimensional signals sampled at high frequencies. While some approaches process raw waveforms [28], most adopt spectrograms with the Fourier Transform [12]. Recent advances employ Variational Autoencoders to discretize waveforms into fixed-granularity tokens for conditioned generation tasks [7,8,25].

Symbolic Representation encodes discrete musical elements with precise, albeit non-uniform, temporal structures. MIDI, as the dominant protocol for real-time performance data, remains the industry standard among musicians and producers. Beyond direct MIDI processing [18], structured representations such as REMI [19], OctupleMIDI [35], and Compound Word [16] are designed to

compress sequence length and enhance information density. Recently ABC notation [31,32,33,29] has gained traction in MuLM communities due to its easy-to-read structure and natural compatibility with natural language encoders.

2.2 Music Understanding Language Model

Recent advances in multimodal language models [22] have enabled new paradigms for music understanding. The key factor in MuLMs is how musical representations (waveform or symbolic), are integrated into LLMs.

Waveform-based MuLMs primarily follow two approaches. The first leverages vector-quantized variational autoencoders to discretize audio signals into tokens [8,34,7,3], achieving high-fidelity reconstruction but requires intensive training. The second employs acoustic feature adapters, mapping waveform features (e.g., MERT [21], CLAP [9]) into LLMs-either via direct projection [10] or cross-attention mechanisms [23].

Symbolic-based MuLMs employ two main tokenization strategies: creating a custom tokenizer, or adopting the pre-trained LM's text tokenizer. For the first approach, [35] use on-off or duration-based representations, while MuPT [29] customizes a tokenizer specifically for ABC notation. Nonetheless, these methods require training from scratch. The second approach processes musical notations as secondary languages through existing LLM vocabularies, exemplified by ChatMusician [33].

While discrete token or adapter approaches can technically combine waveform and symbolic encoders, the scarcity of co-occurring multi-representation training data induces *modality missing* scenarios. UniMuLM tackles this via a unified embedding space construction. Inspired by modality-binding [15,24], CLaMP3 [30], waveform-symbolic proximity enables latent space regularization, allowing single-modality training to enhance cross-modal understanding and support robust inference with partial inputs.

3 Problem Formulation

Our task is to generate a target output $\mathbf{y}$ given music inputs $\mathbf{m}^{\text{inputs}}$ and a task prompt $\mathbf{x}$ (e.g., *"Describe the music in detail"* for music captioning). In this work, $\mathbf{S}$ denotes symbolic notation (ABC), while $\mathbf{W}$ denotes waveform representation. We consider $\mathbf{m} = \{\mathbf{m}_{\text{S}}, \mathbf{m}_{\text{W}}\}$, where these representations are optional and may coexist, describing either identical or distinct music segments. Formally, we define a language token set $\mathcal{T}$. The prompt $\mathbf{x} \in \mathcal{T}^{l_x}$, output $\mathbf{y} \in \mathcal{T}^{l_y}$, and symbolic input $\mathbf{m}_{\text{S}} \in \mathcal{T}^{l_m}$ are all text token sequences. The waveform input $\mathbf{m}_{\text{W}} \in \mathbb{R}^{s \cdot r}$ is a sequence of audio samples with duration s and sampling rate r. Following standard language modeling, we frame this as an autoregressive estimation:$P(\mathbf{y}_i \mid \mathbf{x}, \mathbf{m}^{\text{inputs}}, \mathbf{y}_{1:i-1})$.

4 Methodology

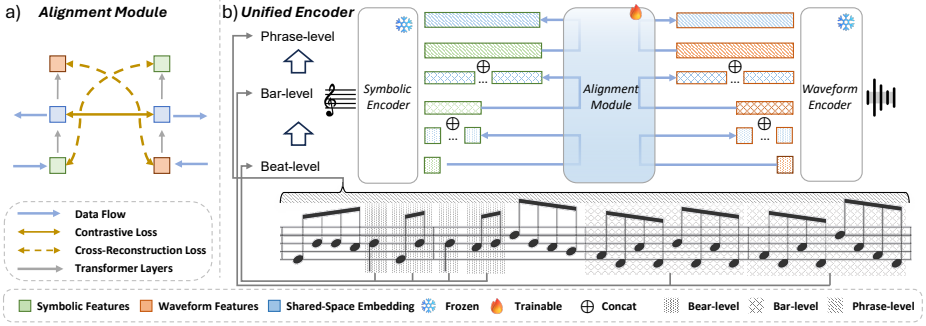

Fig. 3: The illustration of Unified Encoder. (a) The Alignment Module is trained using contrastive loss and cross-reconstruction loss, aligning symbolic feature and waveform features into a shared embedding space; (b) The hierarchical structure is presented, where features at different granularities are interconnected, aiming to enhance the model's information extraction capabilities.

As Figure 4 shows, the UniMuLM framework employs a unified encoder architecture designed to process diverse music representations with a decoder-only transformer serving as the backbone, as illustrated. During training, we first train the encoder and then jointly optimize the adapter and LLM parameters.

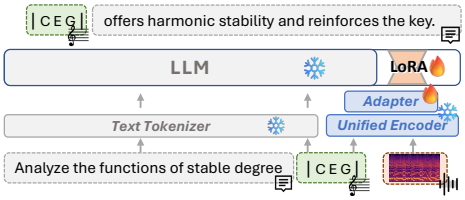

Fig. 4: The overall framework of UniMuLM.

4.1 Unified Encoder

To resolve inherent incompatibilities
between music representations in existing systems, we propose a unified encoder architecture that establishes a shared embedding space while preserving modality-specific features, as Figure 3 shown.

Overview. We address the structural gap between continuous audio signals, which involve continuous temporal sampling without explicit segmentation, and discrete symbolic representations, which organize into irregular and semantically defined units, as mentioned in Section 2, by using musically informed units as patch boundaries to maintain tempo alignment. Conventional fixed-duration audio windowing often misaligns with symbolic timing. To balance temporal resolution and structural consistency, we introduce a three-level hierarchical encoding at the beat, bar, and phrase levels. Larger temporal units improve computational efficiency, while finer segmentation preserves local detail. Our multi-scale design employs learnable attention to enable dynamic interaction between granularities, with task-specific effects evaluated in Section 5.

In this paper, we define $\Gamma = \{\text{beat}, \text{bar}, \text{phrase}\}$, where $\gamma \in \Gamma$ denotes an arbitrary granularity, and $\Psi = \{\text{symbolic}, \text{waveform}\}$, where $\psi \in \Psi$ denotes a music representation, with $\bar{\psi}$ representing the complementary one.

Implementation. The unified encoder processes symbolic and waveform inputs through distinct yet interconnected pathways. For waveform inputs ($\mathbf{m}_{\text{W}}$), we employ EnCodec [7] to generate latent representations $\mathbf{E}^{\gamma}_{\text{W}} \in \mathbb{R}^{l_W \times d}$, where r_c denotes the frame rate and d the latent dimension size. For symbolic inputs ($\mathbf{m}_{\text{S}}$), we tokenize ABC notation at the character level. Features across granularity levels are structured as $\mathbf{E}^{\gamma}_{\text{S}} \in \mathbb{R}^{l_S \times d}$, where l_S is the sequence length, with embedding dimension consistent with waveform representations.

The alignment module integrates dual encoding streams with cross-modal reconstruction. $\text{Enc}^{\gamma}_{\psi}$ and $\text{Dec}^{\gamma}_{\psi}$ are 3-layer Transformers that respectively map input $\mathbf{E}'^{\gamma}_{\psi}$ into shared-space $\mathbf{Z}^{\gamma}_{\phi}$, and decode them into reconstructed embeddings $\hat{\mathbf{E}}^{\gamma}_{\bar{\psi}}$ of the complementary modality. This design enables rich cross-modal interaction while preserving core information in the shared latent space.

The encoder input $\mathbf{E}'^{\gamma}_{\psi}$ differs from $\mathbf{E}^{\gamma}_{\psi}$ through hierarchical information propagation, where higher-level embeddings incorporate lower-level context: [1]

$$\text{Beat-level:} \quad \mathbf{E}'^{\text{beat}}_{\psi} = \mathbf{E}^{\text{beat}}_{\psi}, \tag{1}$$

$$\text{Bar-level:} \quad \mathbf{E}'^{\text{bar}}_{\psi} = \mathbf{E}^{\text{bar}}_{\psi} + \text{Concat}\big(\{\mathbf{Z}^{\text{beat}}_{\psi}\}_n\big), \qquad n \in \{2, 3, 4\}, \tag{2}$$

$$\text{Phrase-level:} \quad \mathbf{E}'^{\text{phrase}}_{\psi} = \mathbf{E}^{\text{phrase}}_{\psi} + \text{Concat}\big(\{\mathbf{Z}^{\text{bar}}_{\psi}\}_n\big), \qquad n = 4. \tag{3}$$

4.2 Adapting Large Language Model

Text encoding. We utilize the pre-defined embedding table from the LM to encode instruction tokens (e.g., prompt $\mathbf{x}$) via the look-up function LM-Emb : $\mathcal{T} \rightarrow \mathbb{R}^{d_T}$, where d_T denotes the dimensionality of the textual embedding. Specifically, $\mathbf{Z}_{\mathbf{x}} = \text{LM-Emb}(\mathbf{x}) \in \mathbb{R}^{l_x \times d}$ represents the transformed embeddings, having the same length l_x as the input tokens. It is worth noting that ABC inputs will also be included as part of text input, following ChatMusician [33], which makes ABC have a dual representation.

Music encoding. Both symbolic ABC $\mathbf{m}^{\gamma}_{\text{S}}$ and waveform inputs $\mathbf{m}^{\gamma}_{\text{W}}$ are encoded by the Unified Encoder into sequences of embeddings $\{\mathbf{Z}^{\gamma}_{\text{S}}\}$ and $\{\mathbf{Z}^{\gamma}_{\text{W}}\}$. For waveform processing under default settings, we adopt a $4/4$ time signature assumption with BPM=120, resulting in beat-level segmentation every 0.5 seconds and bar-level segmentation every 2 seconds.

As we define in the problem formulation section, the LM backbone takes multi-modal embeddings to generate a sequence of textual tokens, expressed as:

$$\text{P}(\mathbf{y}_i \mid \mathbf{x}, \mathbf{m}, \mathbf{y}_{1:i-1}) = \text{LM}\Big(\mathbf{Z}_{\mathbf{x}, \mathbf{y}_{1:i-1}}, \text{A}^{\gamma}_{\text{S}}(\mathbf{Z}^{\gamma}_{\text{S}}), \text{A}^{\gamma}_{\text{W}}(\mathbf{Z}^{\gamma}_{\text{W}})\Big) \tag{4}$$

[1] At the Bar-level, n represents the number of beats per bar, which depends on the time signature.

where the $\mathrm{A}_{\psi}^{\gamma}$ is a fully connected layer that adapts music embeddings to the embedding space and dimensionality of the LM.

4.3 Training Strategy

To mitigate data scarcity of multimodal co-occurrences, we propose a three-stage training strategy: 1) Music Representation Aligning (aligning symbolic and waveform music), 2) Knowledge Injecting (aligning symbolic music and text), 3) Multi-Task Fine-tuning (using waveform tasks to align all modalities).

Stage 1 (Music Representation Aligning). During training, paired training data $\mathcal{D}_{\text{align}} = \{(\mathbf{m}_{\mathrm{S}}^{i,\gamma}, \mathbf{m}_{\mathrm{W}}^{i,\gamma})\}_{i=1}^{n}$ is synthesized by rendering symbolic sequences with randomized instrumental timbres. To align the symbolic and waveform intermediate embeddings within the shared latent space, we apply InfoNCE [14] as contrastive loss:

$$\mathcal{L}_{\text{InfoNCE}}^{\gamma,\psi,(i)} = -\log \frac{\exp(\cos(\mathbf{Z}_{\psi,i}^{\gamma}, \mathbf{Z}_{\bar{\psi},i}^{\gamma})/\tau)}{\sum_{k=1}^{N} \exp(\cos(\mathbf{Z}_{\psi,i}^{\gamma}, \mathbf{Z}_{\bar{\psi},k}^{\gamma})/\tau)} \tag{5}$$

where $\cos(\cdot,\cdot)$ represents cosine similarity, τ is a temperature parameter, and N is the number of negative samples. To mitigate excessive information loss, we apply a cross-reconstruction loss, represented as: $\mathcal{L}_{\text{rec}}^{\mathrm{S},\gamma} = \|\hat{\mathbf{E}}_{\mathrm{S}} - \mathbf{E}_{\mathrm{S}}\|_2^2$, $\mathcal{L}_{\text{rec}}^{\mathrm{W},\gamma} = \|\hat{\mathbf{E}}_{\mathrm{W}}^{\gamma} - \mathbf{E}_{\mathrm{W}}^{\gamma}\|_2^2$. Thus, the loss for representation alignment, which combines both contrastive and reconstruction losses, is denoted as:

$$\underset{\Theta_{\text{Dec, Enc}}}{\operatorname{argmin}} \mathcal{L} = \sum_{\gamma \in \Gamma} \left(\mathcal{L}_{\text{InfoNCE}}^{\gamma} + \mathcal{L}_{\text{rec}}^{\mathrm{S},\gamma} + \mathcal{L}_{\text{rec}}^{\mathrm{W},\gamma} \right) \tag{6}$$

Stage 2 (Knowledge Injecting). We begin by using music knowledge and symbolic music datasets to warm up the pre-trained LLM. During this stage, the unified encoders remain inactive. The training dataset incorporates a mixture of symbolic music represented in ABC notation format, which is processed purely as textual data. Training is achieved through a negative log-likelihood (NLL) objective, where the model predicts the next token $\mathbf{y}_i$ in the sequence based on the previous tokens $\mathbf{y}_{1:i-1}$:

$$\underset{\Theta_{\text{LoRA}}}{\operatorname{argmin}} \mathcal{L} = -\frac{1}{l_{\mathbf{y}}} \sum_{i=1}^{l_{\mathbf{y}}} \log \mathrm{P}(\mathbf{y}_i \mid \mathbf{x}, \mathbf{y}_{1:i-1}) \tag{7}$$

Stage 3 (Multi-Task Fine-tuning). In the final stage, we freeze the Unified Encoder, LoRA-tune the LLM and train adapters to accommodate musical representations for all downstream tasks across different datasets that include symbolic music, waveform music, and textual instructions. We formally present the final stage training as follows:

$$\underset{\Theta_{\text{LoRA},\mathrm{A}_{\mathrm{W}},\mathrm{A}_{\mathrm{S}}}}{\operatorname{argmin}} \mathcal{L} = -\frac{1}{l_{\mathbf{y}}} \sum_{i=1}^{l_{\mathbf{y}}} \log \mathrm{P}(\mathbf{y}_i \mid \mathbf{x}, \mathbf{m}, \mathbf{y}_{1:i-1}) \tag{8}$$

5 Experiments

To comprehensively evaluate UniMuLM's performance and effectiveness, we conducted experiments on four music understanding benchmarks and explored its applicability to two downstream tasks: melody completion and music knowledge answering. We also performed detailed model studies and ablations to analyze the contributions of different components and configurations.

5.1 Experimental Settings

Hyperparameter settings. We employ Llama3-8B as the LM backbone with a hidden dimension of 4096, a learning rate of 5e-6, and a total batch size of 16 across 4 devices, using a 64-rank LoRA with $\alpha = 16$. The Unified Encoder is followed by lightweight adapter modules consisting of a self-attention layer and an MLP, which encode each music feature into a 4096-dimensional embedding aligned with Llama3's hidden size.

Baselines. Waveform-based methods involve GPT-4o [1], LTU [13], Audio-Flamingo [20], LLark [11] and Mu-LLaMA [23]; Symbolic-based methods involve ChatMusician [33] and MuPT [29].[2]

Datasets. We categorize datasets into four core tasks, as summarized in Table 1. (1) Music Knowledge (MK) targets theoretical understanding, drawing on a corpus derived from MusicPile [33] and evaluated with MusicTheoryBench (MTB), a multiple-choice benchmark on composition theory and ABC notation. (2) Music Understanding (MU) aims to bridge waveform music and text, integrating multi-modal datasets: MusicCaps, SongDescriber [26], and MidiCaps [27] for music descriptions, together with MusicQA [23] for complex reasoning. SongDescriber is used only for evaluation, and MIDI sequences in MidiCaps are synthesized into waveforms. (3) Melody Completion (MC) focuses on symbolic generation, while (4) Melody Transcription (MT) involves converting waveform into symbolic notation. Both tasks are constructed from MelodyHub [31]'s ABC melodies at bar-, phrase-, and melody-level granularity, and support both multiple-choice and open-ended generation formats.

Table 1: Summary of training datasets.

Task	Datasets Source	Sampled
Music Knowledge (MK)	MusicPile	200K
Music Understanding (MU)	MusicCaps, MidiCaps, SongDescriber, MusicQA	25K
Melody Completion (MC)	MelodyHub	160K
Melody Transcription (MT)	MelodyHub	125K

5.2 Performance Evaluation

In addition to utilizing BLEU and ROUGE-L (R-L) metrics to evaluate waveform music understanding in an open-ended manner across four datasets, we

[2] To keep the manuscript concise yet self-contained, we moved the detailed description of baselines to our online appendix `https://tuteng0915.github.io/UniMuLM`

Table 2: Performance comparison on music understanding tasks. Best in **bold**, second best underlined.

Category	Model	MusicCaps			SongDescriber			MidiCaps			MusicQA		
		BLEU	R-L	Acc	BLEU	R-L	Acc	BLEU	R-L	Acc	BLEU	R-L	Acc
Baseline	GPT-4o	0.280	0.310	0.875	0.262	0.292	0.830	**0.305**	**0.312**	**0.895**	0.230	0.315	**0.955**
	LTU	0.216	0.248	0.805	0.222	0.237	0.805	0.201	0.223	0.780	0.242	0.328	0.905
	Audio-Flamingo	0.221	**0.320**	0.810	0.218	0.302	0.825	0.213	0.297	0.830	0.234	0.337	0.900
	LLark	0.278	0.250	0.770	0.243	0.237	0.730	0.248	0.268	0.810	0.201	0.194	0.935
	Mu-LLaMA	0.281	0.316	0.780	0.278	0.313	0.710	0.271	0.306	0.735	**0.306**	**0.466**	0.925
UniMuLM	Beat-level	0.217	0.205	0.830	0.204	0.234	0.700	0.190	0.224	0.810	0.212	0.333	0.880
	Bar-level	0.275	0.310	**0.890**	0.290	**0.341**	0.875	0.263	0.311	0.875	0.295	0.409	0.910
	Phrase-level	**0.281**	0.317	0.855	**0.287**	0.329	0.835	0.241	0.287	0.850	0.271	0.369	0.900

also created a 4-option closed-ended evaluation. This was achieved by pairing ground-truth answers with three randomly selected distractors from other music samples (200 test samples per dataset). Taken together, the combination of these perspectives provides a more holistic and reliable assessment of model capability, offering complementary insights into performance. The overall results are summarized in Table 2.

Among baseline models, Mu-LLaMA delivers competitive results on MusicCaps (BLEU: 0.281, R-L: 0.316) and achieves the highest BLEU (0.306) and R-L (0.466) on MusicQA. GPT-4o, in contrast, attains the top scores on MidiCaps (BLEU: 0.305, R-L: 0.312, Acc: 0.895) and also leads in MusicQA accuracy (0.955). Within the UniMuLM variants, the Bar-level encoding achieves the best results on SongDescriber (R-L: 0.341, Acc: 0.875) and maintains strong performance on MusicQA (R-L: 0.409), whereas the Phrase-level encoding slightly surpasses Bar-level on MusicCaps (BLEU: 0.281, R-L: 0.317) and on SongDescriber BLEU (0.287). Overall, Bar-level and Phrase-level encodings exhibit comparable and consistently superior performance over Beat-level, particularly in tasks requiring long-range musical context. By contrast, baseline models demonstrate dataset-specific peaks but display greater variability across BLEU and ROUGE-L metrics.

5.3 Downstream Tasks

The music understanding capabilities of UniMuLM enable it to perform well on various downstream tasks. In this study, we evaluate two categories of downstream tasks: Music knowledge answering and Melody Completion as shown in Table 3 and Table 4.

Music knowledge answering. On the MusicTheoryBench, in Table 3, UniMuLM with bar-level encoding achieves the highest accuracy (0.460), outperforming both general-purpose waveform-based models such as GPT-4o (0.441) and symbolic-focused models like ChatMusician (0.354). This suggests that UniMuLM effectively integrates structural and symbolic music features, enabling more accurate reasoning over musical knowledge. The comparison among different encoding granularities further reveals that bar-level encoding provides the best balance

Table 3: Evaluation on MTB.

Category	Model	ACC
General	GPT-4o	0.441
	Llama3-8B	0.398
MuLM	ChatMusician	0.354
	Mu-LLaMA	0.256
	LTU	0.319
UniMuLM	Beat-level	0.401
	Bar-level	**0.460**
	Phrase-level	0.411

Table 4: Performance on melody continuation and inpainting tasks.

Category	Model	Continuation					Inpainting				
		Acc	Valid	RC	BLEU	R-L	Acc	Valid	RC	BLEU	R-L
General	GPT-4o	0.586	0.912	0.645	0.341	0.556	0.330	**0.963**	0.255	0.122	0.262
	Llama3-8B	0.502	0.756	0.457	0.205	0.213	0.312	0.799	0.120	0.114	0.121
MuLM	MuPT	-	0.798	0.385	0.272	0.295	-	-	-	-	-
	ChatMusician	0.553	0.852	0.630	0.487	0.532	0.454	0.885	0.121	0.069	0.082
UniMuLM	Beat-level	**0.688**	0.942	0.618	0.497	0.623	0.611	0.955	0.327	**0.133**	0.243
	Bar-level	0.673	**0.952**	0.644	**0.502**	**0.643**	**0.615**	0.948	**0.323**	0.128	**0.238**
	Phrase-level	0.668	0.947	**0.650**	0.495	0.638	0.611	0.965	0.322	0.128	0.222

between local rhythmic detail and global structural coherence, that appears crucial for representing hierarchical concepts in music theory.

Melody completion. We evaluate melody completion using two complementary subtasks: Continuation, which extends a given melodic fragment, and Inpainting, which reconstructs missing segments, in Table 4. Performance is assessed via both discrete-choice accuracy (Acc) and open-ended generation metrics, including BLEU, R-L, Rhythmic Consistency (RC) for pitch-agnostic temporal alignment, and Validity for syntactic correctness in ABC notation (e.g., consistent beat count). Across both subtasks, UniMuLM consistently outperforms baseline models. The bar-level variant demonstrates the strongest overall results, achieving, for example, 0.673 Acc in Continuation and 0.625 Acc in Inpainting. It also attains the highest BLEU and R-L scores while preserving high Validity and RC, indicating that it maintains rhythmic integrity without sacrificing melodic structure. Notably, the beat-level variant trails the other two granularities, further highlighting the importance of capturing longer-range musical dependencies.

5.4 Ablation and Model Analysis

We systematically evaluate our framework by (1) examining how the unified encoder design impacts Stage 1 effectiveness; (2) verifying if unified encoder outperforms existing encoders under identical experimental settings, and (3) assessing whether Stage 2 knowledge injection as warm-up training benefits Stage 3 LLM performance; (4) exploring the interaction synergies among tasks. Additional analyses and extended results are also provided in the online appendix.

Unified Encoder Design. To validate the Unified Encoder design, we conduct systematic ablation studies focusing on three critical components: (1) contrastive objective selection (InfoNCE, replaced in ablation with cosine similarity), (2) the cross-reconstruction object, and (3) inter-granularity concatenation. We evaluate cross-format retrieval ability, where S2W denotes retrieving a waveform representation from a symbolic query, and W2S denotes retrieving a symbolic representation from a waveform query. Retrieval accuracy is reported as hit@1 among 10 candidates, serving as a proxy for cross-modal alignment capability. As shown in Table 5, all three components are essential: removing any one results in a substantial drop in retrieval accuracy. Removing the InfoNCE objective or

Table 5: Ablation study on unified encoder designs.

Config	InfoNCE Loss ↓			Rec. Loss ↓			S2W Retrieval ↑			W2S Retrieval ↑		
	Beat	Bar	Phrase	Beat	Bar	Phrase	Beat	Bar	Phrase	Beat	Bar	Phrase
Proposed	0.14	0.15	0.18	0.11	0.24	0.35	0.94	0.93	0.97	0.89	0.88	0.93
w/o InfoNCE	-	-	-	0.13	0.19	0.42	0.90	0.89	0.82	0.83	0.85	0.88
w/o Recon.	0.25	0.23	0.21	-	-	-	0.43	0.38	0.27	0.38	0.33	0.44
w/o Concat	0.18	0.17	0.21	0.15	0.29	0.44	0.82	0.73	0.90	0.80	0.77	0.85

Table 6: Comparison with existing encoders and Ablation of Stage 2

Encoder	MusicCaps	SongDescriber	MidiCaps	MusicQA	MTB
MERT	0.310	0.275	0.287	0.322	0.368
CLAP	0.305	0.316	0.259	0.288	0.354
CLaMP2	0.297	0.310	0.241	0.302	0.385
Proposed	0.310	0.341	0.311	0.409	0.463
- w/o Stage2	0.281	0.323	0.282	0.378	0.356

inter-granularity concatenation lead to moderate but consistent declines in retrieval—for instance, while eliminating the cross-reconstruction objective causes severe performance degradation with S2W accuracy plunging to as low as 0.27 for Phrase-level and W2S dropping to 0.33–0.44 across granularities, confirming its pivotal role in enabling thorough bidirectional information fusion.

Comparison with Existing Encoders. We report ROUGE-L scores on four music understanding benchmarks and the accuracy on MusicTheoryBench, comparing our proposed encoder with MERT [21], CLAP [9], and CLaMP2 [32] (Table 6). Our model achieves the highest scores across all benchmarks, with especially large gains on knowledge-intensive datasets such as MusicQA (+0.087) and MusicTheoryBench (+0.078). These improvements indicate a stronger and more consistent integration of symbolic and audio representations.

Effect of Stage 2 Knowledge Injection. Stage 2 pretraining serves to inject symbolic-music-specific knowledge before Stage 3 LLM adaptation. Skipping Stage 2 leads to modest degradation on SongDescriber (−0.018), but much sharper declines on knowledge-sensitive benchmarks like MusicQA (−0.031) and MusicTheoryBench (−0.107), as shown in Table 6. This indicates that Stage 2 is particularly beneficial for tasks requiring deeper reasoning over musical form and theory, whereas more descriptive tasks can partially rely on Stage 3 adaptation alone, and the knowledge injection phase plays a crucial role in bridging symbolic structure with downstream reasoning ability.

5.5 Multi-Task Synergy

This experiment quantifies cross-task interactions by evaluating four core music-related tasks under different training configurations: Music Knowledge Modeling (MK, Accuracy on MusicTheoryBench), Music Understanding (MU, R-L across

four benchmarks), Music Captioning (MC, multiple-choice accuracy), and Music Transcription (MT, open-ended generation F1), as summarized in Table 7.

Table 7: Multi-task synergy evaluation.

Tasks	Metrics			
	MK (Acc)	MU (R-L)	MC (Acc)	MT (F1)
MK	0.454	–	–	–
MK, MU	0.441	0.324	–	–
MK, MU, MC	0.457	0.338	0.652	–
All	**0.460**	**0.342**	**0.673**	**0.303**

The results demonstrate several clear trends. First, adding MU to MK slightly decreases knowledge accuracy (-0.014), suggesting that shared representations introduce a trade-off between factual precision. Second, incorporating MC further improves both MK and MU, and additionally provides strong captioning accuracy (0.652), highlighting that tasks requiring semantic grounding benefit from multi-modal supervision. Finally, training on all four tasks achieves the highest overall scores across all metrics, indicating that a fully joint setup promotes complementary learning rather than interference.

6 Conclusion

We propose UniMuLM, a unified framework that bridges symbolic and waveform music within LLMs, using a novel encoder to coordinate multi-granularity features and maintain structural integrity while mitigating temporal inconsistencies across different music representation. To train UniMuLM efficiently and effectively, we implement a multi-stage optimization strategy, enabling the model to process diverse musical inputs and achieve competitive performance across various music-related tasks. This work shifts the paradigm from single-representation reliance to synergistic multi-representation integration, advancing music understanding and enriching the landscape of multimodal language models.

Limitations. UniMuLM currently has three main limitations. First, the alignment module was only trained on single-track, single-instrument synthesized music, primarily within Western tonal and metrical conventions; non-Western traditions and non-isochronous rhythms are out of scope. Second, due to computational constraints, our use of 4-bit quantization and LoRA-based fine-tuning limits the model's capacity. This is a trade-off for efficiency and may yield non-leading performance compared to full-parameter optimization. Third, UniMuLM's output is limited to text and ABC notation because it relies on a text-based LLM decoder. As a result, it does not directly generate MIDI or waveforms.

Future work. Future research may focus on three directions: First, enhancing the alignment training by integrating multi-track symbolic parsing with acoustic scene analysis in combination with real recordings and synthesizer timbres. Second, developing a hybrid decoder that generates both symbolic and acoustic tokens. This will allow for direct MIDI generation using event-based tokens [19,16] and waveform synthesis using audio codecs [7,34].

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., *et al.*: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Agostinelli, A., Denk, T.I., Borsos, Z., Engel, J.H., Verzetti, M., Caillon, A., *et al.*: Musiclm: Generating music from text. CoRR abs/2301.11325 (2023)
3. Borsos, Z., Marinier, R., Vincent, D., Kharitonov, E., Pietquin, O., Sharifi, M., *et al.*: Audiolm: A language modeling approach to audio generation. IEEE ACM Trans. Audio Speech Lang. Process. 31, 2523–2533 (2023)
4. Chou, Y., Chen, I., Chang, C., Ching, J., Yang, Y.: Midibert-piano: Large-scale pre-training for symbolic music understanding. CoRR abs/2107.05223 (2021)
5. Chu, Y., Xu, J., Zhou, X., Yang, Q., Zhang, S., Yan, Z., *et al.*: Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. CoRR abs/2311.07919 (2023)
6. Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., *et al.*: Simple and controllable music generation. In: NeurIPS (2023)
7. Défossez, A., Copet, J., Synnaeve, G., Adi, Y.: High fidelity neural audio compression. Trans. Mach. Learn. Res. (2023)
8. Dhariwal, P., Jun, H., Payne, C., Kim, J.W., Radford, A., Sutskever, I.: Jukebox: A generative model for music. CoRR abs/2005.00341 (2020)
9. Elizalde, B., Deshmukh, S., Ismail, M.A., Wang, H.: CLAP learning audio concepts from natural language supervision. In: IEEE ICASSP, pp. 1–5. IEEE (2023).
10. Deng, Z., Ma, Y., Liu, Y., Guo, R., Zhang, G., Chen, W., *et al.*: Musilingo: Bridging music and text with pre-trained language models for music captioning and query response. In: NAACL-HLT (Findings). pp. 3643–3655. Association for Computational Linguistics (2024)
11. Gardner, J., Durand, S., Stoller, D., Bittner, R.M.: Llark: A multimodal foundation model for music. In ICML (2024)
12. Gong, Y., Chung, Y., Glass, J.R.: AST: audio spectrogram transformer. In: Interspeech. pp. 571–575. ISCA (2021)
13. Gong, Y., Luo, H., Liu, A.H., Karlinsky, L., Glass, J.R.: Listen, think, and understand. In: ICLR (2024)
14. Gutmann, M., Hyvärinen, A.: Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: AISTATS. JMLR Proceedings, vol. 9, pp. 297–304. JMLR.org (2010)
15. Girdhar, R., El-Nouby, A., Liu, Z., Singh, M., Alwala, K.V., Joulin, A., Misra, I.: ImageBind: One embedding space to bind them all. In: CVPR (2023)
16. Hsiao, W., Liu, J., Yeh, Y., Yang, Y.: Compound word transformer: Learning to compose full-song music over dynamic directed hypergraphs. In: AAAI. pp. 178–186. AAAI Press (2021)
17. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., *et al.*: Lora: Low-rank adaptation of large language models. In: ICLR (2022)
18. Huang, C.A., Vaswani, A., Uszkoreit, J., Simon, I., Hawthorne, C., Shazeer, N., *et al.*: Music transformer: Generating music with long-term structure. In: ICLR (2019)
19. Huang, Y., Yang, Y.: Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions. In: ACM Multimedia. pp. 1180–1188. ACM (2020)
20. Kong, Z., Goel, A., Badlani, R., Ping, W., Valle, R., Catanzaro, B.: Audio flamingo: A novel audio language model with few-shot learning and dialogue abilities. In: ICML (2024)

21. Li, Y., Yuan, R., Zhang, G., Ma, Y., Chen, X., Yin, H., *et al.*: MERT: acoustic music understanding model with large-scale self-supervised training. In: ICLR (2024)
22. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: NeurIPS (2023)
23. Liu, S., Hussain, A.S., Sun, C., Shan, Y.: Music understanding llama: Advancing text-to-music generation with question answering and captioning. In: ICASSP. pp. 286–290. IEEE (2024)
24. Liu, X., Xia, X., Huang, Z., Chua, T.-S.: Towards Modality Generalization: A Benchmark and Prospective Analysis. CoRR abs/2412.18277 (2024)
25. Liu, X., Tu, T., Ma, Y., Chua, T.-S.: Extending Visual Dynamics for Video-to-Music Generation. CoRR abs/2504.07594 (2025)
26. Manco, I., Weck, B., Doh, S., Won, M., Zhang, Y., Bogdanov, D., *et al.*: The song describer dataset: a corpus of audio captions for music-and-language evaluation. CoRR **abs/2311.10057** (2023)
27. Melechovský, J., Roy, A., Herremans, D.: Midicaps - A large-scale MIDI dataset with text captions. In: ISMIR. pp. 858–865. ISMIR (2024)
28. van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., *et al.*: Wavenet: A generative model for raw audio. In: SSW. p. 125. ISCA (2016)
29. Qu, X., Bai, Y., Ma, Y., Zhou, Z., Lo, K.M., Liu, J., *et al.*: Mupt: A generative symbolic music pretrained transformer. In: ICLR (2025)
30. Wu, S., Guo, Z., Yuan, R., Jiang, J., Doh, S., Xia, G., *et al.*: Clamp 3: Universal music information retrieval across unaligned modalities and unseen languages. In: ACL (Findings). pp. 2605–2625. Association for Computational Linguistics (2025)
31. Wu, S., Wang, Y., Li, X., Yu, F., Sun, M.: Melodyt5: A unified score-to-score transformer for symbolic music processing. In: ISMIR. pp. 642–650. ISMIR (2024)
32. Wu, S., Wang, Y., Yuan, R., Guo, Z., Tan, X., Zhang, G., *et al.*: Clamp 2: Multimodal music information retrieval across 101 languages using large language models. In: NAACL (Findings). pp. 435–451. Association for Computational Linguistics (2025)
33. Yuan, R., Lin, H., Wang, Y., Tian, Z., Wu, S., Shen, T., *et al.*: Chatmusician: Understanding and generating music intrinsically with LLM. In: ACL. pp. 6252–6271. Association for Computational Linguistics (2024)
34. Zeghidour, N., Luebs, A., Omran, A., Skoglund, J., Tagliasacchi, M.: Soundstream: An end-to-end neural audio codec. IEEE ACM Trans. Audio Speech Lang. Process. 30, 495–507 (2022)
35. Zeng, M., Tan, X., Wang, R., Ju, Z., Qin, T., Liu, T.: Musicbert: Symbolic music understanding with large-scale pre-training. In: ACL/IJCNLP (Findings). pp. 791–800. Association for Computational Linguistics (2021)