UniMuMo: Unified Text, Music and Motion Generation

Han Yang[1], **Kun Su**[2], **Yutong Zhang**[3], **Jiaben Chen**[4], **Kaizhi Qian**[5], **Gaowen Liu**[6], **Chuang Gan**[4]

[1]The Chinese University of Hong Kong,
[2]University of Washington,
[3]The University of British Columbia
[4]University of Massachusetts Amherst,
[5]MIT-IBM Watson AI Lab,
[6]Cisco Research

Abstract

We introduce UniMuMo, a unified multimodal model capable of taking arbitrary text, music, and motion data as input conditions to generate outputs across all three modalities. To address the lack of time-synchronized data, we align unpaired music and motion data based on rhythmic patterns to leverage existing large-scale music-only and motion-only datasets. By converting music, motion, and text into token-based representation, our model bridges these modalities through a unified encoder-decoder transformer architecture. To support multiple generation tasks within a single framework, we introduce several architectural improvements. We propose encoding motion with a music codebook, mapping motion into the same feature space as music. We introduce a music-motion parallel generation scheme that unifies all music and motion generation tasks into a single transformer decoder architecture with a single training task of music-motion joint generation. Moreover, the model is designed by fine-tuning existing pre-trained single-modality models, significantly reducing computational demands. Extensive experiments demonstrate that UniMuMo achieves competitive results on all unidirectional generation benchmarks across music, motion, and text modalities.

Code — https://github.com/hanyangclarence/UniMuMo
Website —
https://hanyangclarence.github.io/unimumo_demo/
Extended version — https://arxiv.org/abs/2410.04534

Introduction

Music and body movements are synchronized and inseparable. The beat and metrical structures in rhythm encourage the spontaneous coordination of body motion with music (Large 2000), activating the motor-related areas of human brains (Keller and Rieger 2009). Dance particularly exemplifies this connection through choreography that aligns with the music's rhythm, melody and emotion. Meanwhile, even though most people are not professional musicians or dancers, they often interpret music and dance using simple, natural language. This descriptive text serves as a vital bridge between understandable ideas and abstract concepts in music and motion.

The synergy between music, motion, and text provides a natural motivation to create a model capable of understanding and creating contents across all these modalities. Moreover, building a framework that can flexibly generate music, motion, and text in arbitrary combinations is crucial for real-world applications, even though existing models already achieve impressive results in unidirectional generation tasks such as text-to-music (Copet et al. 2023), music-to-motion (Tseng, Castellon, and Liu 2023), motion-to-music (Zhu et al. 2022a) and motion-to-text (Jiang et al. 2023). In the real world, there is a demand for diverse generative abilities, and more complex generation tasks may be necessary, such as creating dance sequences based on both music and textual descriptions. Training individual models for each unique combination, although potentially yielding better output quality, would significantly increase training costs, deployment efforts and storage requirements. Thus, a unified model that supports all combinations of conditioning and generation tasks, rather than a collection of separate models or training adapters to incorporate individual models, offers a more cost-effective solution. To this end, we introduce a novel task of dynamically generating music, motion, and text in a multitude of combinations unifiedly. As demonstrated in Fig. 1, this task is designed to handle diverse generative scenarios, ranging from text-to-music, text-to-motion, to more complex combinations like text-to-music-plus-motion or music-plus-text-to-motion.

However, the task could be challenging, especially in two aspects: i) the lack of comprehensive datasets that include all three modalities - music, motion, and text - limits the development of a general and unified model. While there are individual datasets for music-only (Santana et al. 2020), motion-only (Mahmood et al. 2019), music to motion (Li et al. 2021b) and text to motion (Guo et al. 2022a), a holistic and large-scale dataset that encompasses all three modalities still remains absent; ii) designing a unified architecture that supports both the conditioning and generation of all three modalities is challenging, mainly due to the significant differences between the neural representations for the three modalities and the multiplicity of desired generation tasks.

To address the first challenge of lacking paired data, we propose to align unpaired music and motion sequences based on their rhythmic patterns. Specifically, we extract both music beats and motion visual beats, then employ dynamic time

Figure 1: UniMuMo is able to perform generation tasks on any combination of music, motion, and text. The tasks shown in the figure include text-to-aligned-music-motion, music-to-motion, motion-to-music, music-captioning, and motion-captioning.

warping to find the alignment and warp the motion sequence to adjust the motion visual beats to match the music beats. We found that such augmentation is accurate and efficient. With the augmented synchronized music-motion data, we can utilize existing music and motion datasets to train our unified generative model. Additionally, we construct text descriptions from music and motion metadata using a mixture of template filling, large language model generation and music-based language model generation, striking a balance between diversity, language fluency and description accuracy.

To overcome the second challenge, we propose a novel framework, UniMuMo, to unify the generation of different modalities. Our pipeline consists of three main stages: a music-motion joint tokenizer that encodes music and motion sequences into discrete representations within the same space, a music-motion transformer-decoder model trained on the task of music-motion joint generation, and a music-motion captioner that generates text descriptions from music and motion features. In the first stage, we bridge the modality gap between music and motion by mapping motion into the music feature space. Specifically, instead of using separate Vector-Quantized Variational Autoencoders (VQ-VAE) to quantize music and motion sequences, we encode motion with the codebook of a pre-trained music VQ-VAE, namely Encodec (Défossez et al. 2022). This design facilitates the unification of music and motion within the same generative framework in the subsequent stage. In the second stage, we train a unified music and motion generative model with a novel task of music-motion joint generation from text conditions. To enable the mutual conditioning of music and motion, and unlock the music-to-motion and motion-to-music generation capabilities, we introduce a novel music-motion parallel generation scheme, where we perform two mutually conditioned streams of autoregressive generation of aligned music and motion simultaneously. With the reuse of Encodec and joint encoding of motion in the previous stage, the current stage can be effectively achieved by fine-tuning the pre-trained text-to-music model associated with Encodec, namely MusicGen (Copet et al. 2023), equipping it with additional motion conditioning and generation capabilities while maintaining its music generation capabilities. In the third stage, we fine-tune a T5 decoder for music and motion captioning tasks, using the features extracted by the music-motion decoder trained in stage 2. To transform the decoder into an effective feature extractor, we replace its causal self-attention layers with trainable full self-attention layers, and fine-tune them together with the T5 decoder on music and motion captioning tasks. Extensive experiments demonstrate that UniMuMo achieves competitive performance across all unidirectional generation tasks in music, motion, and text when compared with existing state-of-the-art models, demonstrating the effectiveness and versatility of our approach.

Our work offers significant advancements in multimodal generative research, summarized as follows:

- To the best of our knowledge, this is the first unified framework capable of arbitrarily generating content across music, motion, and text.
- To address the shortage of paired multimodal data, we augment and enrich existing large-scale datasets with music-motion data alignment and text augmentations.
- We propose a novel joint codebook for encoding music and motion sequences, along with a music-motion parallel generation scheme, facilitating multiple generation tasks within a single architecture.
- Our framework achieves results comparable to SOTAs across all generation tasks in music, motion, and text.

Related Work

Text to Music. Text-conditioned music generation has been widely studied in recent years. There are two main branches: diffusion-based and transformer-based. For diffusion-based models, Riffusion (Forsgren and Martiros 2022) uses a latent text-to-image diffusion model to generate spectrograms, which are then converted into audio clips; Mousai (Schneider, Jin, and Schölkopf 2023) proposes training a diffusion model in the latent space of a diffusion autoencoder; Noise2Music (Huang et al. 2023a) introduces a cascade of diffusion models that first generates the audio in a coarse form and then progressively refine it. AudioLDM (Liu et al. 2023a)

proposes to train a latent diffusion model using CLAP (Wu et al. 2023) embeddings, a language-audio joint representation, for text conditioning. For transformer-based models, MusicLM (Agostinelli et al. 2023) proposes to encode music into high-level "semantic tokens" and low-level "acoustic tokens", and use a cascade of transformer decoders to generate the two levels stage by stage. MusicGen (Copet et al. 2023) leverages a single-stage transformer decoder to model the hierarchical music tokens directly.

Music to Text. Several models have been proposed for audio captioning. WAC (Kadlčík et al. 2023) proposes to transfer a pre-trained speech-to-text Whisper model to the music captioning task. LTU (Gong et al. 2023) takes the concatenated music embeddings and text embeddings as input to a large language model and directly trains caption generation using language modeling objectives. LP-MusicCaps (Doh et al. 2023) uses a transformer encoder-decoder structure, where the music spectrogram is first encoded by the encoder and then cross-attended by the decoder for text generation. MU-LLaMA (Liu et al. 2023b) leverages a frozen LLaMA (Touvron et al. 2023) and fine-tunes a Music Understanding Adapter to fuse music features into the LLaMA model.

Music to Motion. Most of the works on music-conditioned dance generation are based on transformers. Several approaches (Li et al. 2021a; Fan et al. 2022; Pu and Shan 2022) adopt similar structures that first use a music transformer encoder and a motion transformer encoder to encode music and initial motion into representations separately, and then employ a transformer decoder for cross-modal fusion and motion generation. Bailando (Siyao et al. 2022) proposes to train a transformer on motion features encoded by a choreographic memory module, which is the codebook of a motion VQ-VAE. Besides autoregressive transformers, EDGE (Tseng, Castellon, and Liu 2023) adopts a transformer-based diffusion model capable of both dance generation and editing.

Motion to Music. Most of the relevant works focus on generating corresponding music from video input. Foley Music (Gan et al. 2020) focuses on generating music for videos of people playing instruments, and uses Musical Instrument Digital Interface (MIDI) to bridge the gap between body key points and the final music. Similarly, RhythmicNet (Su, Liu, and Shlizerman 2021) extends the scenarios to arbitrary motion videos by first estimating visual rhythm and conditionally generating drum and piano music. Dance2Music (Aggarwal and Parikh 2021) encodes a dance similarity matrix with CNN and predicts the next note with an LSTM autoregressively. CDCD (Zhu et al. 2022b) proposes a single-stage method that uses a discrete latent diffusion model to generate music spectrograms conditioned on video features. D2M-GAN (Zhu et al. 2022a) proposes a GAN-based model to generate the music tokens based on video and pose features.

Text-Music-Motion Aligned Data Generation

To model arbitrary generation across music, motion, and text, we propose to expand existing music and motion datasets by aligning motion with music and synthesizing textual descriptions. The data generation pipeline includes four major steps: 1) music beat detection, 2) visual beat detection, 3) music-motion alignment, and 4) text description synthesis.

Music Beat Detection. We estimate music beats from a music waveform $Y \in \mathbb{R}^{T_w}$, where T_w represents the number of samples, using a Bidirectional-LSTM-based model from (Chiu, Su, and Yang 2021). This model performs beat tracking on extracted drum features and non-drum features separately, then aggregates the results with a learnable fuser. We manually evaluate the accuracy of this beat tracking model and find that it performs well in most test cases, outperforming the beat tracking methods in the Librosa API (McFee et al. 2015). The resulting music beats are represented as a binary sequence $B_m \in \mathbb{R}^{T_w}$, where each frame is marked as 'beat' or 'non-beat.'

Visual Beats Detection. Given a 3D motion sequence $M \in \mathbb{R}^{T_m \times J \times 3}$ where T_m represents the number of frames, J the number of joints, and the last dimension indicates x, y, z coordinates, we obtain visual beats in three steps. In the first stage, we calculate the motion directogram (Davis and Agrawala 2018), a 2D matrix that factors motion into different motion angles, similar to how an audio spectrogram factors sound amplitude into different frequencies. Specifically, we first compute the first-order difference of the motion sequence $\Delta M_t = M_t - M_{t-1}$. Based on its motion angle, we assign the motion magnitude of every joint into one of the bins in $2\pi/N_{\text{bins}}$. The motion directogram $M_d(t, \theta)$ is obtained by summing the motion magnitudes of each bin: $M_d(t, \theta) = \sum_j \Delta M_t(j) \mathbf{1}_\theta(\angle M_t(j))$, where $\mathbf{1}_\theta(\phi) = 1$ if $|\theta - \phi| \leq 2\pi/N_{\text{bins}}$ else 0. In the second stage, we convert the motion directogram to the kinematic offset M_k, which represents the motion changes, similar to the onset envelope in an audio spectrogram. We first obtain motion flux M_f, which represents the deceleration in various directions, by computing the negative first-order difference of the directogram ΔM_d. We then average each frame of M_f and filter the top 1% peaks to obtain kinematic offset M_k. In the last stage, we use dynamic programming to compute the visual beats by designing an objective function that selects strong visual changes from kinematic offsets and encourages equal-spacing beats. More details can be found in Appendix. The final visual beats are also represented as a binary sequence $B_v \in \mathbb{R}^{T_m}$, where each frame is marked as 'beat' or 'non-beat'.

Music-Motion Alignment. We apply dynamic time warping to determine the optimal matching between music beats B_m and visual beats B_v, finding the alignment even though the duration of these two binary sequences could be different. Finally, we warp motion sequences by interpolating according to the warping curve to obtain aligned music-motion pairs. The reason for warping motion to match music, rather than the reverse, is that music beats tend to be steady, so warping music could result in perceptually unacceptable changes. More details can be found in Appendix.

Text Description Synthesis. To compensate for the absence of text descriptions in our used datasets, we employ two methods for captions synthesis: (1) using Music Understanding Language Model to generate caption directly from audio; and (2) using Large Language Model to synthesize captions from metadata (genre, tempo, *etc.*), striking a balance between

musical accuracy and diversity. Examples and more details are shown in Appendix.

UniMuMo Framework

UniMuMo consists of three training stages to enable arbitrary generation between music, motion, and text. In stage 1, we encode aligned music and motion data into discrete tokens. To efficiently bridge the gap between the two modalities, we propose to use a frozen pre-trained audio tokenizer Encodec (Défossez et al. 2022) and train a motion tokenizer that reuses the same residual codebooks of the audio tokenizer. In stage 2, we fine-tune a state-of-the-art text-to-music transformer decoder (Copet et al. 2023) by conducting the task of generating music and motion tokens simultaneously with music and motion text descriptions. At the inference stage, we can perform parallel generation to unlock applications of music and motion generation. In stage 3, we treat the pre-trained music-motion decoder model in stage two as a feature extractor and fine-tune a T5 decoder on language modeling task for music and motion captioning. An overview of the UniMuMo framework is shown in Figure 2.

Stage 1. Music and Motion Joint Tokenization

While existing tokenization approaches can faithfully reconstruct the music or motion individually, the correlations between the two modalities become intricate in distinct spaces. Therefore, directly applying them in the unified generation framework poses challenges. Besides, a music tokenizer usually requires more training resources and time to achieve high-quality reconstruction than a motion tokenizer. Inspired by these facts, we introduce an efficient and effective way to encode music and motion into a joint latent space. We propose using a pre-trained audio tokenizer, Encodec (Défossez et al. 2022), and training a new motion encoder-decoder. The motion encoder encodes the motion into the same embedding space as the music and reuses the frozen music Residual Vector Quantizers (RVQ) to discretize the motion into tokens. From these tokens, the motion decoder can decode to reconstruct the motion. Given the higher complexity and richer information in music compared to motion, the learned music codebook is theoretically capable of encoding motion.

Specifically, given a waveform $Y \in \mathbb{R}^{T \cdot f_w}$ with T the audio duration and f_w the sample rate, Encodec first encodes it into a continuous tensor of $X_{\text{music}} \in \mathbb{R}^{d \times T \cdot f_r}$, where $f_r \ll f_w$ is the frame rate of the residual codebook and d is the dimension of codebook entries. X_{music} is then quantized by the RVQ into music tokens $Q_{\text{music}} \in \{1, \dots, M\}^{K \times T \cdot f_r}$, where K is the number of RVQ and M is the number of codebook entries. For an aligned motion sequence of the same duration $M \in \mathbb{R}^{d_m \times T \cdot f_m}$ with frame rate f_m and feature dimension d_m, our motion encoder encodes it into $X_{\text{motion}} \in \mathbb{R}^{d \times T \cdot f_r}$, the same shape as X_{music}, which is then tokenized by the same RVQ into motion tokens $Q_{\text{motion}} \in \{1, \dots, M\}^{K \times T \cdot f_r}$. The motion decoder decodes the motion feature after RVQ, resulting in $\hat{M}$. The motion encoder-decoder is trained by minimizing the motion reconstruction loss together with a commitment loss $\mathcal{L}_{\text{commit}}$ from the codebook:

$$\mathcal{L}_{\text{total}} = \frac{1}{|\mathcal{D}|} \sum_{M \in \mathcal{D}} (\|M - \hat{M}\|_2 + \lambda \mathcal{L}_{\text{commit}}) \tag{1}$$

where $\mathcal{D}$ is the motion dataset and λ controls the strength of the commitment loss. Empirically, λ is set to 0.02.

With this design, the music-motion joint tokenization can effectively learn multimodal correlations by mapping motion features into the same space as music, without the need to train another computationally heavy music autoencoder. Moreover, it enables direct use the text-to-music model associated with Encodec as an initialization for the following music-motion decoder model, significantly reducing training costs and enhancing the performance. Experimentally, such feature alignment is crucial to learning the joint generation of music and motion within a single transformer model.

Stage 2. Music and Motion Generation from Text

In this stage, we modify and fine-tune an existing state-of-the-art text-to-music model with the music and motion tokens extracted from Stage 1, enabling it to handle all tasks related to music and motion generation, such as text-to-music-motion and motion-to-music. In particular, we employ MusicGen (Copet et al. 2023), an open-source, single-stage transformer decoder model that can generate multi-level music tokens with a specific codebook interleaving pattern. Following their practice, we apply the delay pattern for both music and motion tokens, utilize a T5 encoder for encoding text descriptions, and adopt cross-attention to incorporate text conditioning features into the transformer decoder.

To enable the autoregressive generation of music and motion within a unified framework, we propose training on the task of music-motion joint generation, together with a novel parallel generation scheme, where two streams (*i.e.*, music and motion) of predict-next-token generation are conducted simultaneously, with each stream conditioned on each other. Specifically, given the music tokens Q_{music} and motion tokens Q_{motion} with the same shape $K \times S$ where $S = T \cdot f_r$ is the sequence length, we first transform them with delay pattern (Copet et al. 2023) into Q'_{music} and Q'_{motion} respectively, resulting shape $K \times S'$, where $S' = S + K - 1$. We then concatenate them in time dimension into Q_{input} of the shape $K \times 2S'$ as the input to the transformer decoder. The model's output is transformed back to the normal pattern for loss calculation. Training on music-motion joint generation, we adopt the predict-next-token objectives for both music and motion tokens in each forward pass:

$$\begin{aligned} \mathcal{L} = -\frac{1}{|\mathcal{D}|} \sum_{Q \in \mathcal{D}} \Bigg\{ & \mu \cdot \sum_{t=1}^{S} \log \mathbb{P}\left[Q_t^{\text{music}} | Q_{<t}^{\text{music}}, Q_{<t}^{\text{motion}} \right] \\ & + (1-\mu) \cdot \sum_{t=1}^{S} \log \mathbb{P}\left[Q_t^{\text{motion}} | Q_{<t}^{\text{music}}, Q_{<t}^{\text{motion}} \right] \Bigg\} \end{aligned} \tag{2}$$

where μ balances between music loss and motion loss, and $\mathbb{P}$ denotes predict-next-token probability of the model. Empirically, μ is set to 0.85. To enable the parallel autoregressive generation, we apply a cross-modal causal attention mask, as

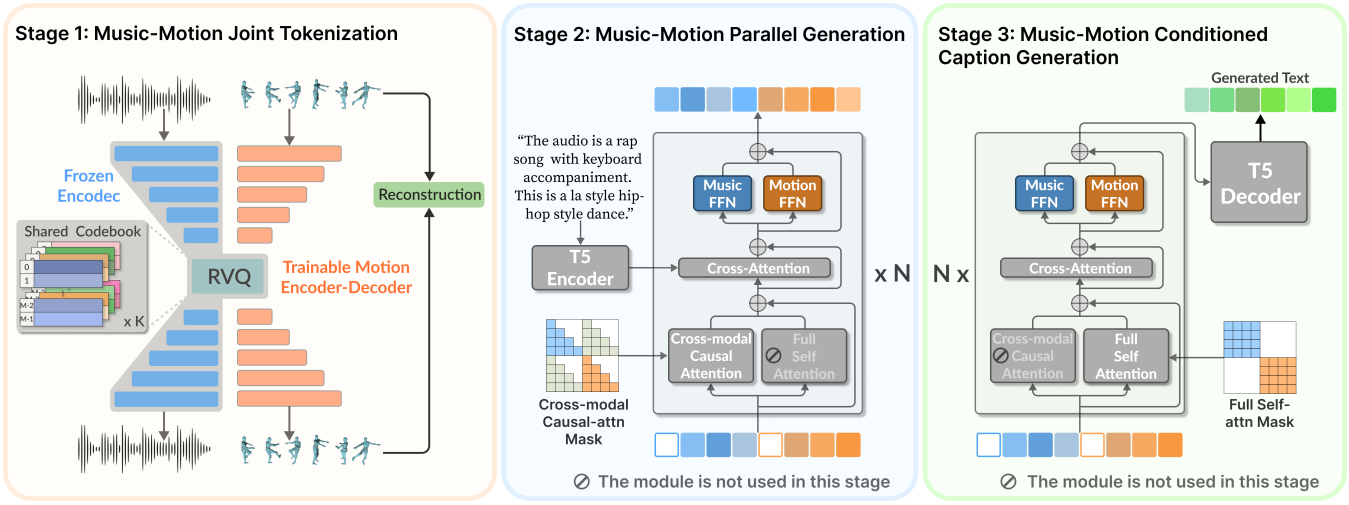

Figure 2: Overview: The training of UniMuMo consists of three stages: In stage 1, we train a motion RVQ-VAE using the frozen codebook from a pre-trained music RVQ-VAE to encode motion into the same space as music. In stage 2, we fine-tune a pre-trained music transformer decoder model on the text-to-music-motion task using the music-motion parallel generation scheme. In stage 3, we fine-tune a T5 decoder for music-motion captioning using the previous music-motion decoder as a feature extractor.

shown in Stage 2 of Figure 2. The causal attention mask is of shape $2S' \times 2S'$, each quarter of which is an $S' \times S'$ lower triangular matrix, allowing music and motion tokens to have both cross-modal and uni-modal causal attention. A further illustration of the strategy can be found in Appendix.

With the above construction, the model can perform parallel sampling during inference, enabling the prediction of the next token for both music and motion concurrently:

$$\hat{Q}_t^{\text{music}} = \underset{i \in M}{\text{argmax}} \mathbb{P}[Q_{t,i}^{\text{music}} | \hat{Q}_{<t}^{\text{music}}, \hat{Q}_{<t}^{\text{motion}}] \quad (3)$$

$$\hat{Q}_t^{\text{motion}} = \underset{i \in M}{\text{argmax}} \mathbb{P}[Q_{t,i}^{\text{motion}} | \hat{Q}_{<t}^{\text{music}}, \hat{Q}_{<t}^{\text{motion}}] \quad (4)$$

where M is the codebook size. With this sampling strategy, we can conduct the joint generation of music and motion under text conditions. Additionally, it facilitates zero-shot music-to-motion and motion-to-music generation. For example, given a music sequence $Q_{1:S}^{\text{music}}$, an aligned motion sequence can be autoregressively sampled by

$$\hat{Q}_t^{\text{motion}} = \underset{i \in M}{\text{argmax}} \mathbb{P}[Q_{t,i}^{\text{motion}} | Q_{<t}^{\text{music}}, \hat{Q}_{<t}^{\text{motion}}] \quad (5)$$

An illustration of the sampling process can also be found in Appendix.

Considering the inherent differences between music and motion, we further introduce the following changes to the pre-trained MusicGen to alleviate the mutual interference between the two modalities. First, we add another trainable embedder for motion tokens, with which the model can learn to differentiate the two modalities. Second, to ensure the temporal parallelism, we add positional encodings $\{E_1, E_2, \ldots, E_{S'}\}$ to music and motion separately, instead of using a holistic positional encoding of length $2S'$. Third, inspired by the idea of Mixture of Experts (MoE), we introduce an additional feed-forward network (FFN) for motion in each transformer layer. As shown in Fig. 2, in each forward pass, the first half of the feature (*i.e.*, music features) is processed by the music FFN, and the second half (*i.e.*, motion features) by the motion FFN. Fourth, we add a new motion classification head at the end of the network to distinguish motion code prediction from music code prediction. Note that for the new modules introduced above, we initialize the motion embedder and FFNs with the corresponding components from the pre-trained MusicGen. With a joint motion VQ-VAE trained in Stage 1, such initialization ensures that music features are not confused by uninitialized motion features at the beginning of training, allowing the music generation capability to be better preserved.

Following MusicGen, text conditioning is added with cross-attention. In the framework of music-motion joint generation, we add the text condition of two modalities independently. We first encode music descriptions and motion descriptions separately into features and apply classifier-free guidance dropout independently. Then, during cross-attention on text conditions, we specialize the attention mask to allow music features to attend only to music conditions and motion features to attend only to motion conditions.

By fine-tuning the model on the music-motion dataset with the above settings, we find that the model learns to generate motion in parallel with music quickly while still keeping its music-generation ability. With a single training task of music-motion joint generation, various applications could be achieved in a zero-shot fashion, including text-to-music, text-to-motion, music-to-motion, motion-to-music, motion-text-to-music, *etc.*.

Models	FAD_{VGG} ↓	KL↓	CLAP↑
Riffusion (Forsgren and Martiros 2022)	14.8	2.06	0.19
Mubert (Mubert-Inc. 2022)	9.6	1.58	-
Mousai (Schneider, Jin, and Schölkopf 2023)	7.5	1.59	0.23
MusicLM (860M) (Agostinelli et al. 2023)	4.0	1.31	-
MusicGen (300M) (Copet et al. 2023)	4.9	1.42	0.27
AudioLDM 2-Full (346M) (Liu et al. 2023a)	**3.13**	**1.17**	**0.38**
Ours (300M)	5.93	1.99	0.27
MusicGen (fine-tuned on our data)	5.81	1.97	0.28
Ours (trained on data with vocals)	4.11	1.95	0.29

Table 1: Comparison of text-to-music generation on MusicCaps. **Bold** and underlined results are the best and second-best results.

Stage 3. Music and Motion Captioning

The final stage is for caption generation, where we treat the fine-tuned music-motion decoder in the previous stage as a feature encoder for music and motion, and fine-tune another T5 decoder to generate captions for music and motion.

However, using the music-motion decoder directly as a feature extractor brings challenges. Firstly, the self-attention in the decoder is done causally, which is inadequate for capturing rich music and motion features. Secondly, since the input of the model is the concatenation of music and motion, we are limited to input music-motion pairs for captioning, which is inflexible.

To address these issues, we introduce a trainable full self-attention module, initialized with the trained cross-modal causal attention module, as shown in Fig. 2, Stage 3. Inspired by BLIP (Li et al. 2022), which claims that the major difference between transformer encoders and decoders lies in the self-attention layers, with embedding layers and FFNs functioning similarly, we therefore fine-tune only the newly introduced full self-attention modules together with the T5 decoder on caption generation task, keeping the rest of the music-motion decoder unchanged. Considering that captions of music and motion are independent, we remove the cross-attention areas on the attention mask.

In practice, we first randomly mask the entire music or motion tokens as empty, and concatenate them together as input Q_{input}. This allows us to conduct music or motion captioning independently. Next, we forward it through the music-motion decoder with a null condition, where full self-attention is applied. We then take the output of the last hidden layer of the model as the feature, which is cross-attended by the T5 text decoder. We fine-tune the model with the language modeling task, and the generation target is either music caption or motion caption, depending on the input masking.

Experiment

Evaluations

We conduct extensive evaluations of our model across various tasks and metrics. More implementation details about hyperparameter choices, dataset, metrics and training/evaluation setups are in Appendix.

Text-to-Music. In Table 1, we compare our UniMuMo with Riffusion (Forsgren and Martiros 2022), Mubert (Mubert-Inc. 2022), Mousai (Schneider, Jin, and Schölkopf 2023), MusicLM (Agostinelli et al. 2023), MusicGen (Copet et al. 2023) and AudioLDM 2 (Liu et al. 2023a). We evaluate the performance on MusicCaps, with results of SOTAs directly sourced from their respective papers. We employ three metrics: Frechet Audio Distance (FAD_{VGG}) (Kilgour et al. 2018), Kullback-Leibler Divergence (KL) (Kreuk et al. 2022) and CLAP similarity (CLAP) (Wu et al. 2023; Huang et al. 2023b). The first two metrics measure the audio quality, while the last one measures the correspondence between generated audio and text descriptions. Note that the audio quality of our model does not match with SOTA models. We argue that this might be due to the poor audio quality of our training data. Following MusicGen, we also use vocal-free training data. To achieve this, we use Demucs (Défossez 2021; Rouard, Massa, and Défossez 2023) to remove the vocal part of the music in Music4All dataset. Nonetheless, we observe that many of the processed audio are of bad quality. This is testified by the experiment of fine-tuning MusicGen on our dataset for the same number of epochs while keeping all other settings the same (*e.g.*, sequence length, batch size). As shown in Table 1, the audio quality of the tuned model also degrades. We also tried training the model on the original dataset with vocals, resulting in improved quantitative scores. However, the generated music is not perceptually good, often filled with weird and meaningless vocals. This phenomenon, where training on music with vocals yields better quantitative scores, is also reported in MusicGen.

Models	Beats Coverage↑	Beats Hit↑
Dance2Music (Aggarwal and Parikh 2021)	83.5	82.4
Foley Music (Gan et al. 2020)	74.1	69.4
CMT (Di et al. 2021)	85.5	83.5
D2M-GAN (Zhu et al. 2022a)	88.2	84.7
CDCD (Zhu et al. 2022b)	**93.9**	**90.7**
Ours	93.0	88.4

Table 2: Comparison of motion-conditioned music generation on AIST++.

Models	$Dist_k$ →	$Dist_g$ →	Beat Align.↑
Real	10.61	7.48	0.24
Bailando (Siyao et al. 2022)	7.92	7.72	0.23
FACT (Li et al. 2021a)	10.85	6.14	0.22
EDGE (Tseng, Castellon, and Liu 2023)	**10.58**	**7.62**	**0.27**
Ours (music conditioned)	10.68	10.35	0.24
Ours (text conditioned)	9.14	9.37	0.25

Table 3: Comparison of music-conditioned and text-conditioned dance generation.

Dance-to-Music. In Table 2, we compare UniMuMo with Dance2Music (Aggarwal and Parikh 2021), Foley Mu-

Models	Bleu↑	Meteor↑	Rouge↑	BertScore↑
LTU (Gong et al. 2023)	0.238	0.250	0.332	0.876
LP-MusicCaps (Doh et al. 2023)	0.165	0.202	0.281	0.879
MU-LLaMA (Liu et al. 2023b)	0.238	**0.354**	**0.475**	**0.913**
Ours	**0.261**	0.291	0.369	0.892

Table 4: Comparison of music captioning on MusicQA dataset.

Methods	R-Precision↑		MMDist↓	Bleu↑		ROUGE-L↑	Cider↑	BertScore↑
	Top1	Top3		@1	@4			
Real	0.506	0.800	2.986	-	-	-	-	-
MotionGPT (Jiang et al. 2023)	**0.534**	0.803	2.978	42.61	6.04	34.47	7.92	31.57
TM2T (Guo et al. 2022b)	0.525	**0.814**	2.995	**61.76**	**21.98**	**47.40**	**71.12**	37.27
Ours	0.520	0.806	**2.958**	52.84	9.27	40.11	6.22	**40.90**

Table 5: Comparison of motion captioning on HumanML3D dataset.

sic (Gan et al. 2020), CMT (Di et al. 2021), D2M-GAN (Zhu et al. 2022a) and CDCD (Zhu et al. 2022b) on dance-conditioned music generation. For evaluation, we adopt Beats Coverage and Beats Hit (Zhu et al. 2022a), both of which measure the alignment of generated music with motion.

Music/Text-to-Dance. In Table 3, we compare UniMuMo's dance-generation capabilities with Bailando (Siyao et al. 2022), FACT (Li et al. 2021a) and EDGE (Tseng, Castellon, and Liu 2023) on AIST++ dataset. We evaluate UniMuMo on both music-conditioned and text-conditioned dance generation tasks. Although there is currently no established benchmark for the text-to-dance task, we can also apply the same evaluation metrics to measure and compare the quality of generated dance. For evaluation metrics, we adopt kinetic distribution spread (Dist_k) and geometric distribution spread (Dist_g) to measure the diversity. Additionally, we employ the beat alignment score to measure the alignment between conditioning audio and generated dance. Following EDGE, we evaluate the motion sequences on 5-second clip. For text-to-dance, we directly evaluate the dance that is jointly generated with music, conditioned on both music and motion captions, and we calculate the beat alignment score between the generated dance and music. The quantitative scores show that UniMuMo achieves competitive results on music-conditioned dance generation, even though it hasn't been fine-tuned on AIST++ music. For text-conditioned generation, it achieves inferior dance quality since there is no ground truth music for reference, but also gains a higher beat alignment score due to the joint generation.

Music-to-Text. In Table 4, we compare UniMuMo against SOTA music captioning models including LTU (Gong et al. 2023), LP-MusicCaps (Doh et al. 2023) and MU-LLaMA (Liu et al. 2023b). The evaluation is conducted on the MusicQA dataset released by (Liu et al. 2023b), which is a music-related question-answering dataset. We take the answers to the question "Describe the audio" together with the corresponding music as evaluation data, totaling 552 music-caption pairs. Following MU-LLaMa, the metrics we use includes Bleu, Meteor, Rouge_L and BertScore, which are all common evaluation metrics in natural language processing.

Motion-to-Text. In Table 5, we compare UniMuMo with TM2T (Guo et al. 2022b) and MotionGPT (Jiang et al. 2023) for motion captioning using the HumanML3D test set. Following MotionGPT, we adopt the motion-retrieval precision (R-Precision) to measure the accuracy of motion-text matching using top-1 and top-3 retrieval accuracy, multi-modal distance (MM Dist) to measure the distance between motion and text, and other popular natural language processing metrics, including Blue, Rouge, Cider and BertScore, to assess the linguistic quality. Since we source only 50% of our training motion data from HumanML3D, and the motion is augmented to align with music beats, UniMuMo still lags behind the best SOTA in certain metrics for HumanML3D motion captioning task.

Based on the quantitative results presented above, UniMuMo achieves competitive performance compared to the SOTA benchmarks across various single-modal generation tasks. Specifically, in the motion-to-music, music-to-motion, music captioning and motion captioning tasks, UniMuMo generally ranks second among the SOTAs. However, in the text-to-music task, UniMuMo's performance is not as competitive, which we argue may be attributed to the limitations in our training data.

Conclusion

In this paper, we introduce UniMuMo, the first unified framework for arbitrary generation across music, motion, and text. To address the limitations of paired multimodal data, we expand existing datasets with rhythm-based music-motion alignment and text augmentation, thus creating a comprehensive new dataset. To build a unified model, we propose novel architectural designs, including a music-motion joint tokenizer for bridging modality gaps and a music-motion parallel generation scheme for synchronized music and motion generation. Extensive experiments show that UniMuMo achieves competitive performance in all unidirectional generative tasks. We believe our framework will not only open up new avenues for multimodal generation but also inspire future advancements in this rapidly evolving field.

References

Aggarwal, G.; and Parikh, D. 2021. Dance2music: Automatic dance-driven music generation. *arXiv preprint arXiv:2107.06252*.

Agostinelli, A.; Denk, T. I.; Borsos, Z.; Engel, J.; Verzetti, M.; Caillon, A.; Huang, Q.; Jansen, A.; Roberts, A.; Tagliasacchi, M.; Sharifi, M.; Zeghidour, N.; and Frank, C. 2023. MusicLM: Generating Music From Text. *ArXiv*, abs/2301.11325.

Chiu, C.-Y.; Su, A. W.-Y.; and Yang, Y.-H. 2021. Drum-aware ensemble architecture for improved joint musical beat and downbeat tracking. *IEEE Signal Processing Letters*, 28: 1100–1104.

Copet, J.; Kreuk, F.; Gat, I.; Remez, T.; Kant, D.; Synnaeve, G.; Adi, Y.; and Défossez, A. 2023. Simple and Controllable Music Generation. *arXiv preprint arXiv:2306.05284*.

Davis, A.; and Agrawala, M. 2018. Visual rhythm and beat. *ACM Transactions on Graphics (TOG)*, 37(4): 1–11.

Défossez, A. 2021. Hybrid Spectrogram and Waveform Source Separation. In *Proceedings of the ISMIR 2021 Workshop on Music Source Separation*.

Défossez, A.; Copet, J.; Synnaeve, G.; and Adi, Y. 2022. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*.

Di, S.; Jiang, Z.; Liu, S.; Wang, Z.; Zhu, L.; He, Z.; Liu, H.; and Yan, S. 2021. Video background music generation with controllable music transformer. In *Proceedings of the 29th ACM International Conference on Multimedia*, 2037–2045.

Doh, S.; Choi, K.; Lee, J.; and Nam, J. 2023. Lp-musiccaps: Llm-based pseudo music captioning. *arXiv preprint arXiv:2307.16372*.

Fan, D.; Wan, L.; Xu, W.; and Wang, S. 2022. A bi-directional attention guided cross-modal network for music based dance generation. *Computers and Electrical Engineering*, 103: 108310.

Forsgren, S.; and Martiros, H. 2022. Riffusion - Stable diffusion for real-time music generation.

Gan, C.; Huang, D.; Chen, P.; Tenenbaum, J. B.; and Torralba, A. 2020. Foley music: Learning to generate music from videos. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, 758–775. Springer.

Gong, Y.; Luo, H.; Liu, A. H.; Karlinsky, L.; and Glass, J. 2023. Listen, Think, and Understand. *arXiv preprint arXiv:2305.10790*.

Guo, C.; Zou, S.; Zuo, X.; Wang, S.; Ji, W.; Li, X.; and Cheng, L. 2022a. Generating Diverse and Natural 3D Human Motions From Text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5152–5161.

Guo, C.; Zuo, X.; Wang, S.; and Cheng, L. 2022b. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In *European Conference on Computer Vision*, 580–597. Springer.

Huang, Q.; Park, D. S.; Wang, T.; Denk, T. I.; Ly, A.; Chen, N.; Zhang, Z.; Zhang, Z.; Yu, J.; Frank, C.; et al. 2023a. Noise2music: Text-conditioned music generation with diffusion models. *arXiv preprint arXiv:2302.03917*.

Huang, R.; Huang, J.; Yang, D.; Ren, Y.; Liu, L.; Li, M.; Ye, Z.; Liu, J.; Yin, X.; and Zhao, Z. 2023b. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models. *arXiv preprint arXiv:2301.12661*.

Jiang, B.; Chen, X.; Liu, W.; Yu, J.; Yu, G.; and Chen, T. 2023. MotionGPT: Human Motion as a Foreign Language. *arXiv preprint arXiv:2306.14795*.

Kadlčík, M.; Hájek, A.; Kieslich, J.; and Winiecki, R. 2023. A Whisper transformer for audio captioning trained with synthetic captions and transfer learning. *arXiv preprint arXiv:2305.09690*.

Keller, P. E.; and Rieger, M. 2009. Musical movement and synchronization. *Music Perception*, 26(5): 397–400.

Kilgour, K.; Zuluaga, M.; Roblek, D.; and Sharifi, M. 2018. Fr\'echet Audio Distance: A Metric for Evaluating Music Enhancement Algorithms. *arXiv preprint arXiv:1812.08466*.

Kreuk, F.; Synnaeve, G.; Polyak, A.; Singer, U.; Défossez, A.; Copet, J.; Parikh, D.; Taigman, Y.; and Adi, Y. 2022. Audiogen: Textually guided audio generation. *arXiv preprint arXiv:2209.15352*.

Large, E. W. 2000. On synchronizing movements to music. *Human movement science*, 19(4): 527–566.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, 12888–12900. PMLR.

Li, R.; Yang, S.; Ross, D. A.; and Kanazawa, A. 2021a. Ai choreographer: Music conditioned 3d dance generation with aist++. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 13401–13412.

Li, R.; Yang, S.; Ross, D. A.; and Kanazawa, A. 2021b. Learn to Dance with AIST++: Music Conditioned 3D Dance Generation. arXiv:2101.08779.

Liu, H.; Tian, Q.; Yuan, Y.; Liu, X.; Mei, X.; Kong, Q.; Wang, Y.; Wang, W.; Wang, Y.; and Plumbley, M. D. 2023a. AudioLDM 2: Learning Holistic Audio Generation with Self-supervised Pretraining. *arXiv preprint arXiv:2308.05734*.

Liu, S.; Hussain, A. S.; Sun, C.; and Shan, Y. 2023b. Music understanding LLaMA: Advancing text-to-music generation with question answering and captioning. *arXiv preprint arXiv:2308.11276*.

Mahmood, N.; Ghorbani, N.; Troje, N. F.; Pons-Moll, G.; and Black, M. J. 2019. AMASS: Archive of Motion Capture as Surface Shapes. In *International Conference on Computer Vision*, 5442–5451.

McFee, B.; Raffel, C.; Liang, D.; Ellis, D. P.; McVicar, M.; Battenberg, E.; and Nieto, O. 2015. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8.

Mubert-Inc. 2022. Mubert. https://mubert. com/, https://github.com/MubertAI/ Mubert-Text-to-Music. Accessed: 2024-03-05.

Pu, J.; and Shan, Y. 2022. Music-driven dance regeneration with controllable key pose constraints. *arXiv preprint arXiv:2207.03682*.

Rouard, S.; Massa, F.; and Défossez, A. 2023. Hybrid Transformers for Music Source Separation. In *ICASSP 23*.

Santana, I. A. P.; Pinhelli, F.; Donini, J.; Catharin, L.; Mangolin, R. B.; Feltrim, V. D.; Domingues, M. A.; et al. 2020. Music4all: A new music database and its applications. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, 399–404. IEEE.

Schneider, F.; Jin, Z.; and Schölkopf, B. 2023. Mo\^ usai: Text-to-Music Generation with Long-Context Latent Diffusion. *arXiv preprint arXiv:2301.11757*.

Siyao, L.; Yu, W.; Gu, T.; Lin, C.; Wang, Q.; Qian, C.; Loy, C. C.; and Liu, Z. 2022. Bailando: 3d dance generation by actor-critic gpt with choreographic memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11050–11059.

Su, K.; Liu, X.; and Shlizerman, E. 2021. How does it sound? *Advances in Neural Information Processing Systems*, 34: 29258–29273.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Roziere, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. CoRR, abs/2302.13971, 2023. doi: 10.48550.

Tseng, J.; Castellon, R.; and Liu, K. 2023. Edge: Editable dance generation from music. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 448–458.

Wu, Y.; Chen, K.; Zhang, T.; Hui, Y.; Berg-Kirkpatrick, T.; and Dubnov, S. 2023. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

Zhu, Y.; Olszewski, K.; Wu, Y.; Achlioptas, P.; Chai, M.; Yan, Y.; and Tulyakov, S. 2022a. Quantized gan for complex music generation from dance videos. In *European Conference on Computer Vision*, 182–199. Springer.

Zhu, Y.; Wu, Y.; Olszewski, K.; Ren, J.; Tulyakov, S.; and Yan, Y. 2022b. Discrete contrastive diffusion for cross-modal music and image generation. In *The Eleventh International Conference on Learning Representations*.