# MotionGPT: Human Motion as a Foreign Language

**Biao Jiang**[1*] **Xin Chen**[2*] **Wen Liu**[2] **Jingyi Yu**[3] **Gang Yu**[2] **Tao Chen**[1†]
[1]Fudan University [2]Tencent PCG [3] ShanghaiTech University
https://github.com/OpenMotionLab/MotionGPT

## Abstract

Though the advancement of pre-trained large language models unfolds, the exploration of building a unified model for language and other multimodal data, such as motion, remains challenging and untouched so far. Fortunately, human motion displays a semantic coupling akin to human language, often perceived as a form of body language. By fusing language data with large-scale motion models,motion-language pre-training that can enhance the performance of motion-related tasks becomes feasible. Driven by this insight, we propose MotionGPT, a unified, versatile, and user-friendly motion-language model to handle multiple motion-relevant tasks. Specifically, we employ the discrete vector quantization for human motion and transfer 3D motion into motion tokens, similar to the generation process of word tokens. Building upon this "motion vocabulary", we perform language modeling on both motion and text in a unified manner, treating human motion as a specific language. Moreover, inspired by prompt learning, we pre-train MotionGPT with a mixture of motion-language data and fine-tune it on prompt-based question-and-answer tasks. Extensive experiments demonstrate that MotionGPT achieves state-of-the-art performances on multiple motion tasks including text-driven motion generation, motion captioning, motion prediction, and motion in-between.

## 1 Introduction

Recent years have witnessed a significant breakthrough in pre-trained large language models such as GPT [36, 37, 3, 28], BERT [7], and T5 [38, 5], which lead to the convergence of language [61, 49], image [35, 52, 21], mesh [57, 27] and mutlimodal [8] modeling. Nevertheless, a general pre-trained model for human motion and language has yet to emerge. This pre-trained motion-language model, capable of supporting numerous motion-relevant tasks through prompts, should benefit diverse fields like gaming, robotics, virtual assistant, and human behavior analysis.

Previous research on human motion has explored various tasks, including motion generation [31, 11, 48, 54, 59], motion captioning [9, 12], and motion prediction [58, 63, 25]. Recent text-to-motion works[48, 60, 32, 54] have attempted to employ pre-trained language-relevant models [7, 35]. For instance, MDM [48] learns a motion diffusion model with conditional text tokens from CLIP [35], while MLD [54] integrates motion latent space to improve the efficiency of motion diffusion process. On the other hand, MotionCLIP [47] and TM2T [12] concentrate on modeling the coupled relationship between motion and text description. However, the above approaches treat motion and language as separate modalities, which often require strictly paired motion and text data. Moreover, since the supervisions are task-specific, they can hardly generalize effectively to unseen tasks or data, as they lack a comprehensive understanding of the relationship between motion and language. We thus focus on building a pre-trained motion-language model, which can generalize to various tasks and learn in-depth motion-language correlation knowledge from more feasible motion and language data.

*These authors contributed equally to this work.
†Corresponding author.

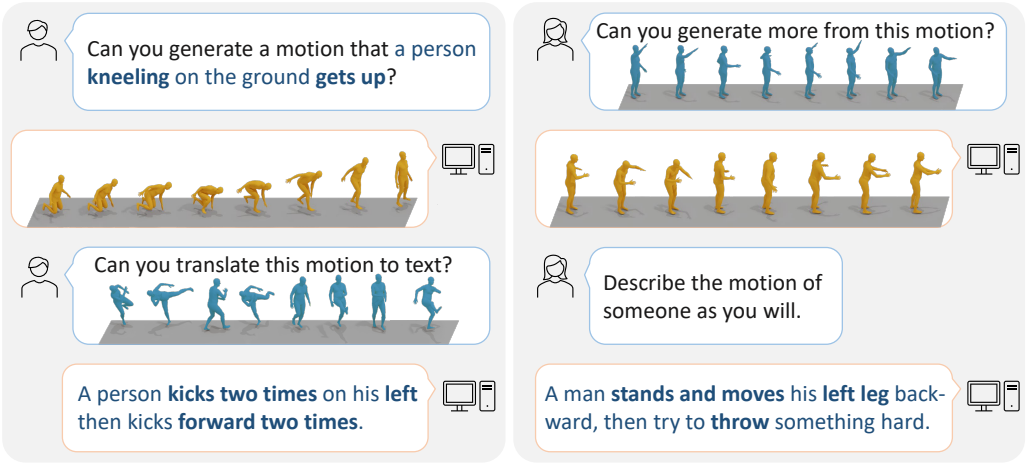



Figure 1: MotionGPT can address diverse motion-relevant tasks uniformly given different instructions. We provide the results on text-to-motion (the upper left), motion captioning (the bottom left), motion completion (the upper right), and the language question-to-answer (the bottom right). The left to right of motion represents the time order. Blue motion denotes the input, and yellow is the generation.

Two challenges are crucial and need to be solved for pre-training a promising motion-language model. The first is modeling the relation between language and motion, and the second is building a uniform multi-task framework that can generalize to new tasks. Fortunately, human motion exhibits a semantic coupling similar to human language, often interpreted as a form of body language. Building upon this observation,we follow vision-language pre-training from BEiT-3 [52] to treat human motion as a specific foreign language. By integrating motion and language data together and encoding them within a single vocabulary, the relationship between motion and language becomes more apparent. Therefore, with recent significantly larger-scale language data and models, the motion-language pre-training has great potential to improve the performance on motion tasks. Meanwhile, this pre-training on language enables textual instructions like prompts in InstructGPT [28] and makes the model more versatile and user-friendly for various motion tasks.

In this work, we propose a uniform motion-language framework, namely MotionGPT, that leverages the strong language generation and zero-shot transfer abilities of pre-trained language models for doing human motion-related tasks. To enable MotionGPT to comprehend and generate human-like motions, we first learn a motion-specific vector quantized variational autoencoder (VQ-VAE) model to construct "motion vocabulary", akin to English vocabulary and then convert raw motion data into a sequence of motion tokens. These tokens are then processed by a pre-trained language model [38, 5] that learns the underlying grammar and syntax of the motion language, as well as its relationship with the corresponding textual descriptions. To effectively integrate language and motion in MotionGPT, we design a two-stage training scheme. We first pre-train the language model on the raw motion dataset to learn the basic grammar and syntax of the motion language. For prompt tuning, we fine-tune the language model on an instruction dataset, which contains both textual descriptions and motion data, to learn the correlation between the two modalities. Extensive experiments demonstrate that MotionGPT achieves state-of-the-art performance on text-to-motion, motion-to-text, motion prediction, and motion in-between.

We summarize our contributions as follows: (1) We propose a uniform motion-language generative pre-trained model, MotionGPT, which treats human motion as a foreign language, introduces natural language models into motion-relevant generation, and performs diverse motion tasks with a single model. (2) We introduce a motion-language training scheme with instruction tuning, to learn from task feedback and produce promising results through prompts. (3) We propose a general motion benchmark for multi-task evaluation, wherein MotionGPT achieves competitive performance across diverse tasks, including text-to-motion, motion-to-text, motion prediction, and motion in-between, with all available codes and data.

| Methods | Text-to-Motion | Motion-to-Text | Motion Prediction | Motion In-between | Random Motion | Random Description |
|---|---|---|---|---|---|---|
| T2M-GPT [48] | ✔ | ✗ | ✗ | ✗ | ✔ | ✗ |
| MLD [54] | ✔ | ✗ | ✗ | ✗ | ✔ | ✗ |
| TM2T [12] | ✔ | ✔ | ✗ | ✗ | ✗ | ✗ |
| MDM [48] | ✔ | ✗ | ✔ | ✔ | ✔ | ✗ |
| MotionDiffuse[60] | ✔ | ✗ | ✔ | ✔ | ✔ | ✗ |
| MotionGPT (Ours) | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |

Table 1: Comparison of recent state-of-the-art methods on diverse motion-relevant tasks. *Random Motion* and *Random Caption* represent unconstrained generation of motions and motion descriptions.

## 2 Related Work

**Human Motion Synthesis** involves generating diverse and realistic human-like motion using multi-modal inputs, such as text [11, 32, 60, 48, 12, 1, 18], action [31, 13, 48, 54], and incomplete motion [58, 63, 25, 48]. Text-to-motion is one of the most important motion generation tasks, due to the user-friendly and convenient language input. MDM [48] proposes a diffusion-based generative model [15] separately trained on several motion tasks. MLD [54] advances the latent diffusion model [45, 40] to generate motions based on different conditional inputs. T2M-GPT [59] investigates a generative framework based on VQ-VAE and Generative Pre-trained Transformer (GPT) for motion generation. Motion completion task generates motion conditioning on partial motions, such as classical motion prediction [58, 63, 25] or motion in-between [48], which generates the intermediate motion while the first and last parts are fixed. Although they show promising results in various human motion tasks, most above methods are limited in using a single model to handle multiple tasks. We thus propose a uniform approach that treats human motion as a foreign language, and leverages the strong language generation and zero-shot transfer abilities of pre-trained language models

**Human Motion Captioning.** To describe human motion with natural languages, [46] learns the mapping from motions to language relying on two statistical models. Furthermore, recurrent networks have also been used in [56, 34]. More recently, TM2T [12] proposed a new motion representation that compresses motions into a short sequence of discrete variables, then uses a neural translation network to build mappings between two modalities. While previous research like TM2T [12] incorporated captioning modules into their training pipeline for motion generation, these approaches are constrained to bidirectional translation between text and motion within one uniform framework.

**Language Models and Multi-Modal.** Large-scale language models (LLMs) [7, 6, 38, 3, 61, 49], enabled by extensive datasets and model size, have demonstrated impressive comprehension and generation capabilities, elevating natural language processing to new heights. BERT [7] pre-trains deep bidirectional language representations that can support downstream tasks. T5 [38] introduced a unified framework that converts all text-based language problems into a text-to-text format. More recent research [53, 2, 28, 5] find that by fine-tuning pre-trained models using input-output pairs consisting of instructions and coupled answers, the performance of pre-trained models can be further improved. FLAN [5] presents an instruction-tuning technique that surpasses the performance of non-tuned models in unseen tasks. Recently, the wave of multi-modal models [21, 16, 20] is intriguing to process text along with other modalities, such as images [21, 16, 8], audio [14, 8], and videos [55]. CLIP [35] further learns a semantic latent representation that couples images with corresponding language descriptions. Despite the success of language models in various vision-language tasks, the development of multi-modal language models that can handle human motion is still limited.

**Motion Language Pre-training.** Existing text-to-motion generation methods [11, 32, 48, 12, 1, 18] can be characterized as caption-to-motion, where the models take in a pure text description of the desired motion. While these methods can generate motions from textual descriptions, they are often limited in supporting instructions from users like InstructGPT [28]. In other words, they do not allow users to provide context-specific instructions for certain applications. MotionCLIP [47] utilizes the language and visual understanding of CLIP [35] to align its latent space with a motion auto-encoder. Meanwhile, many language models, such as T5[38] and InstructGPT [28], have been developed to address diverse language processing tasks, including translation, question answering, and classification. These models are typically designed to map a given text input to a target output, such as a translation or answer. However, while these models have shown remarkable performance in

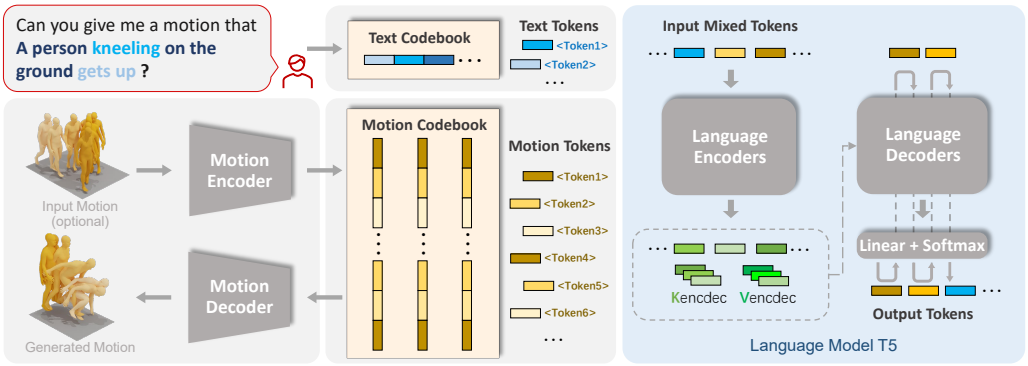



Figure 2: Method overview: MotionGPT consists of a motion tokenizer $\mathcal{V}$ (Sec. 3.1) and a motion-aware language model (Sec. 3.2). Combining *Motion Tokens* learned by $\mathcal{V}$ and *Text Tokens* by text tokenizer, we then learn motion and language jointly utilizing language model as backbone.

language tasks, they have not been widely applied to motion tasks. Therefore, we propose MotionGPT to enable the effective integration of natural language models with human motion tasks, providing a unified solution for motion synthesis problems.

# 3 Method

To involve large language data and models in the motion generation tasks, we propose a unified motion-language framework named MotionGPT. As illustrated in Fig. 2, MotionGPT consists of a motion tokenizer responsible for converting raw motion data into discrete motion tokens (Sec. 3.1), as well as a motion-aware language model that learns to understand the motion tokens from large language pre-training models by corresponding textual descriptions (Sec. 3.2). To address motion-relevant tasks, we introduce a three-stage training scheme (Sec. 3.3) of MotionGPT for the training of motion tokenizer, motion-language pre-training, and instruction tuning.

We first propose the motion tokenizer consisting of a motion encoder $\mathcal{E}$ and a motion decoder $\mathcal{D}$, to encode a $M$ frame motion $m^{1:M} = \{x^i\}_{i=1}^{M}$ into $L$ motion tokens $z^{1:L} = \{z^i\}_{i=1}^{L}, L = M/l$, and decode $z^{1:L}$ back into the motion $\hat{m}^{1:M} = \mathcal{D}(z^{1:L}) = \mathcal{D}(\mathcal{E}(m^{1:M}))$, where $l$ denotes the temporal downsampling rate on motion length. Then, given an $N$ length sentence $w^{1:N} = \{w^i\}_{i=1}^{N}$ describing a motion-related question or demand, MotionGPT aims to generate its answer as $L$ length tokens $\hat{x}^{1:L} = \{\hat{x}^i\}_{i=1}^{L}$. It could be the human motion tokens $\hat{x}_m^{1:L}$ or the text tokens $\hat{x}_t^{1:L}$, which results in a motion $\hat{m}^{1:M}$ or a sentence $\hat{w}^{1:L}$ like a description of the given motion.

## 3.1 Motion Tokenizer

To represent motion in discrete tokens, we pre-train a 3D human motion tokenizer $\mathcal{V}$ based on the Vector Quantized Variational Autoencoders (VQ-VAE) architecture used in [50, 44, 12, 59]. Our motion tokenizer consists of an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$. The encoder generates discrete motion tokens with high informative density, while the decoder is able to reconstruct the motion tokens into motion sequences $\hat{m}^{1:M}$. This approach enables us to efficiently represent motion as a language, facilitating the integration of motion and language for various motion-related tasks.

Specifically, the motion encoder $\mathcal{E}$ first applies 1D convolutions to given frame-wise motion features $m^{1:M}$ along the time dimension, to obtain latent vectors $\hat{z}^{1:L} = \mathcal{E}(m^{1:M})$. Next, we transform $\hat{z}$ into a collection of codebook entries $z$ through discrete quantization. The learnable codebook $Z = \{z^i\}_{i=1}^{K} \subset \mathbb{R}^d$ consists of $K$ latent embedding vectors, each of dimension $d$. The process of quantization $Q(\cdot)$ replaces each row vector $b$ with its nearest codebook entry $b_k$ in $Z$, written as

$$z_i = Q(\hat{z}^i) := \arg\min_{z_k \in Z} \|\hat{z}_i - z_k\|_2 . \tag{1}$$

After quantization, the motion decoder $D$ project $z^{1:L} = \{z^i\}_{i=1}^{L}$ back to raw motion space as the motion $\hat{m}^{1:M}$ with $M$ frames. To train this motion tokenizer, we follow [12, 59] to utilize three

distinct loss functions for training and optimizing the motion tokenizer: $\mathcal{L}_{\mathcal{V}} = \mathcal{L}_r + \mathcal{L}_e + \mathcal{L}_c$, where the reconstruction loss $\mathcal{L}_r$, the embedding loss $\mathcal{L}_e$, and the commitment loss $\mathcal{L}_c$. To further improve the generated motion quality, we follow [59] to utilize L1 smooth loss and velocity regularization in the reconstruction loss, as well as exponential moving average (EMA) and codebook reset techniques [39] to enhance codebook utilization during training. We provide more details about the architecture and the training of our motion tokenizer in the supplement.

### 3.2 Motion-aware Language Model

Employing this motion tokenizer, a human motion $m^{1:M}$ can be mapped to a sequence of motion tokens $z^{1:L}$, allowing for joint representation with similar vocabulary embedding in language models [19, 38, 28]. By combining them in the unified vocabulary, we then learn motion and language jointly. We first represent motion tokens $z^{1:L}$ as a sequence of indices $s^{1:L} = \{s^i\}_{i=1}^{L}$, where $s^i$ corresponds to the index number of motion tokens $z^{1:L}$. On the other hand, previous language models, such as T5 [38], encode text as WordPiece tokens. They utilized a vocabulary of $K_t$ word pieces and trained the SentencePiece [19] model on a mixture of language datasets.

Most previous text-to-motion [12, 54, 59] or motion-to-text [12] approaches employ different modules to handle text and motion individually, while we aim to model text and human motion together and in the same way. To achieve this, we combine the original text vocabulary $V_t = \{v_t^i\}_{i=1}^{K_t}$ with motion vocabulary $V_m = \{v_m^i\}_{i=1}^{K_m}$, which is order-preserving to our motion codebook $Z$. Moreover, $V_m$ includes several special tokens like boundary indicators, for example, </som> and </eom> as the start and end of the motion. Thus, we employ a new unified text-motion vocabulary $V = \{V_t, V_m\}$, and can formulate diverse motion-related tasks in a general format, where both input "words" and output "words" are from the same $V$. These "words" can represent natural language, human motion, or even a mixture of two, depending on the specific task to be solved. Therefore, our MotionGPT allows for the flexible representation and generation of diverse motion-related outputs within a single model.

To address the conditioned generation task, we employ a transformer-based model based on the architecture proposed in [38], which effectively maps the input sequences to the output. Our source input consists of a sequence of tokens $X_s = \{x_s{}^i\}_{i=1}^{N}$, where $x_s \in V$ and $N$ represents the input length. Similarly, the target output is $X_t = \{x_t{}^i\}_{i=1}^{L}$, where $x_t \in V$ and $L$ denotes the output length. As shown in Fig. 2, the source tokens are fed into the transformer encoder, and the subsequent decoder predicts the probability distribution of the potential next token at each step $p_\theta(x_t \mid x_s) = \prod_i p_\theta\left(x_t^i \mid x_t^{<i}, x_s\right)$ in an autoregressive manner. Therefore, during the training process, the objective is to maximize the log-likelihood of the data distribution:

$$\mathcal{L}_{LM} = -\sum_{i=0}^{L_t-1} \log p_\theta\left(x_t^i \mid x_t^{<i}, x_s\right). \tag{2}$$

By optimizing this objective, MotionGPT learns to capture the underlying patterns and relationships from the data distribution, facilitating the accurate and meaningful generation of the target "words". During the inference process, the target tokens are sampled recursively from the predicted distribution $p_\theta\left(\hat{x_t}^i \mid \hat{x_t}^{<i}, x_s\right)$ until the end token (i.e., </s>). This sampling strategy enables the generation of the target sequence in a step-by-step manner, where each token is probabilistically determined based on the previously generated tokens and the given source input.

### 3.3 Training Strategy

Since T5s have only been exposed to language data, represented within a text vocabulary $V_t$, we thus bridge motion and language and enable this language model to comprehend human motion concepts, by learning the motion vocabulary $V_m$. As shown in Fig. 3, our training scheme includes three stages: (1) Training of motion tokenizer, which focuses on learning the motion codebook to represent human motion as discrete tokens. (2) Motion-language pre-training stage, which includes unsupervised and supervised objectives to learn the relationship between motion and language. (3) Instruction tuning stage, which tunes the model based on prompt-based instructions for different motion-relevant tasks.

**Training of Motion Tokenizer.** We first learn the motion tokenizer using the objective defined in Equation 3.1. This training process allows any human motion sequence $\hat{x}^{1:L}$ to be represented as a sequence of motion tokens, enabling seamless integration with textual information. Once optimized, the motion tokenizer remains unchanged throughout the subsequent stages of the pipeline.

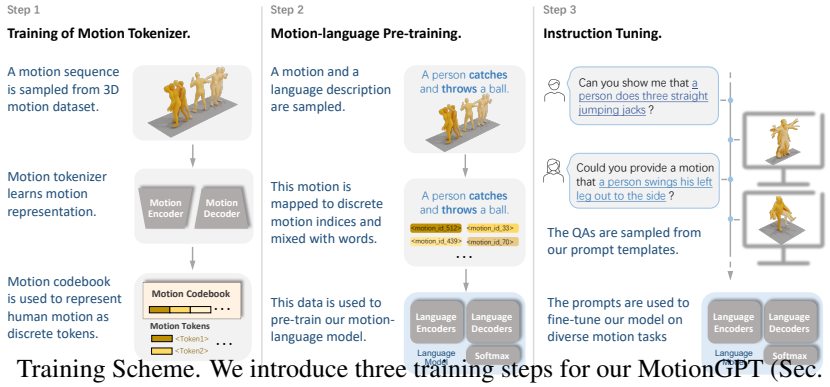



Figure 3: Training Scheme. We introduce three training steps for our MotionGPT (Sec. 3.3): First $\mathcal{V}$ learn a codebook for discrete motion representation. Then we train language using a mixture of language and motion data to learn the semantic coupling between text and motion. Finally, we fine-tune the model in a multi-task text-motion dataset with instructions.

**Motion-language Pre-training Stage.** The T5 models [38, 5] are trained and fine-tuned on natural language datasets with instruction-based phrasing [5, 28]. We continue to pre-train this model using a mixture of language and motions data in both unsupervised and supervised manners: 1) To generalize to various downstream tasks like [7, 37, 38, 28], we follow [38] to design an objective, where a certain percentage (15%) of tokens in the input tokens $X_s$ are randomly replaced with a special sentinel token. On the other side, the corresponding target sequence is constructed by extracting the dropped-out spans of tokens, delimited by the same sentinel tokens used in the input sequence, along with an additional sentinel token to indicate the end of the target sequence. 2) We then learn the motion-language relation by the supervision of paired text-motion datasets [11, 33]. We train MotionGPT on the supervised motion-language translation, where the input is either a human motion or a text description.After unsupervised and supervised training processes, we aim to equip our model with the understanding of text and motion relationships.

**Instruction Tuning Stage.** We construct a multi-task text-motion dataset by formulating it as instructions, building upon the foundation of existing text-to-motion datasets such as HumanML3D [11] and KIT [33]. Specifically, we define 15 core motion tasks, such as motion generation with text, motion captioning, motion prediction, and others. For each task, we compose dozens of different instruction templates, resulting in more than one thousand different tasks, each having a unique instruction prompt. For example, an instruction prompt for motion generation task could be "Can you generate a motion sequence that depicts 'a person emulates the motions of a waltz dance'?". Similarly, for the motion captioning task, the instruction prompt could be "Provide an accurate caption describing the motion of <motion_tokens>", where <motion_tokens> represents a sequence of motion tokens generated by our motion tokenizer. We have demonstrated the efficacy of instruction tuning in Sec. 4.3, which leads to improvement across various tasks and enhances the model performance for unseen tasks or prompts. More examples of prompts are provided in the supplements.

# 4 Experiments

Extensive comparisons evaluate the performance of our MotionGPTs across multiple motion-relevant tasks and datasets. Details of the dataset settings, evaluation metrics, and implementation specifics (Sec. 4.1) are provided. We first present a uniform benchmark by comparing our approach with other SOTAs across various tasks (Sec. 4.2). Then, we evaluate each specific comparison on text-to-motion (Sec. 4.2), motion-to-text (Sec. 4.2), motion prediction and motion in-between (Sec. 4.2). The supplements include more qualitative results, user studies, and further implementation details.

## 4.1 Experimental Setup

**Datasets.** General motion synthesis can support diverse task settings, and thus previous datasets and a modified benchmark are utilized to evaluate MotionGPT. The study primarily focuses on two text-to-motion datasets: HumanML3D [11] and KIT [33]. The KIT dataset provides 6,353 textual descriptions corresponding to 3,911 motion sequences, while the HumanML3D dataset [11] is a more

| Methods | Text-to-Motion | | | Motion-to-Text | | | Motion Prediction | | Motion In-between | |
|---|---|---|---|---|---|---|---|---|---|---|
| | R TOP1↑ | FID↓ | DIV→ | R TOP3↑ | Bleu@4↑ | Cider↑ | FID↓ | DIV→ | FID↓ | DIV→ |
| Real | $0.511^{\pm.003}$ | $0.002^{\pm.000}$ | $9.503^{\pm.065}$ | 0.828 | - | - | 0.002 | 9.503 | 0.002 | 9.503 |
| MLD [54] | $0.481^{\pm.003}$ | $0.473^{\pm.013}$ | $9.724^{\pm.082}$ | - | - | - | - | - | - | - |
| T2M-GPT [48] | $0.491^{\pm.003}$ | $\mathbf{0.116}^{\pm.004}$ | $9.761^{\pm.081}$ | - | - | - | - | - | - | - |
| TM2T [12] | $0.424^{\pm.017}$ | $1.501^{\pm.003}$ | $8.589^{\pm.076}$ | 0.823 | 7.00 | 16.8 | - | - | - | - |
| MDM [48] | $0.320^{\pm005}$ | $0.544^{\pm.044}$ | $9.559^{\pm.086}$ | - | - | - | 6.031 | 7.813 | 2.698 | 8.420 |
| MotionGPT (Ours) | $\mathbf{0.492}^{\pm.003}$ | $0.232^{\pm.008}$ | $\mathbf{9.528}^{\pm.071}$ | **0.827** | **12.47** | **29.2** | **0.905** | **8.972** | **0.214** | **9.560** |

Table 2: Comparison of four motion-related tasks on HumanML3D [11] dataset. The evaluation metrics are computed using the encoder introduced in [11]. The empty columns of previous methods indicate that they can not handle the task. The arrows (→) indicate that closer to *Real* is desirable. **Bold** and underline indicate the best and the second best result on text-to-motion task.

recent dataset that contains 14,616 motion sequences obtained from AMASS [26], along with 44,970 sequence-level textual descriptions. To evaluate MotionGPT as a uniform framework on tasks, such as motion prediction and motion completion (in-between), we utilize the motion sequences available in HumanML3D, which is also a subset of the larger AMASS dataset. Following the previous works [11, 54, 48], we adopt the same motion representation for fair comparisons, which combines joint velocities, positions, and rotations. By using this consistent representation, MotionGPT enables the availability to support further studies in the field. ($cf.$ supplement for the benchmark details.)

**Evaluation Metrics** are summarized as four parts. (1) Motion quality: Frechet Inception Distance (FID) is our primary metric based on a feature extractor [11] to evaluate the distance of feature distributions between the generated and real motions. For motion completion, we utilize metrics used in motion prediction studies [58, 63, 25], such as Average Displacement Error (ADE) and Final Displacement Error (FDE), to evaluate the accuracy of the predicted motion. (2) Generation diversity: We utilize the Diversity (DIV) metric to assess the motions diversity, which calculates the variance through features extracted from the motions [11]. MultiModality (MM) measures the diversity of generated motions within the same text description of motion. (3) Text matching: Based on the feature space from [11], the motion-retrieval precision (R Precision) evaluates the accuracy of matching between texts and motions using Top 1/2/3 retrieval accuracy. Multi-modal Distance (MM Dist) measures the distance between motions and texts. (4) Linguistic quality: We follow [12] utilizing linguistic metrics from natural language studies, including BLUE [29], Rouge [24], Cider [51], and BertScore [62] to evaluate the quality of generated motion captions.

**Implementation Details.** We set the codebook of motion tokenizer as $K \in \mathbb{R}^{512\times512}$ for most comparisons. The motion encoder $\mathcal{E}$ incorporates a temporal downsampling rate $l$ of 4. We utilize T5 [38] as the underlying architecture for our language model, with a baseline model consisting of 12 layers in both the transformer encoder and decoder. The feed-forward networks have an output dimensionality of $d_{\text{ff}} = 3072$, and the attention mechanisms employ an inner dimensionality of $d_{\text{kv}} = 64$. The remaining sub-layers and embeddings have a dimensionality of $d_{\text{model}} = 768$. Moreover, all our models employ the AdamW optimizer for training. The motion tokenizers are trained utilizing a $10^{-4}$ learning rate and a 256 mini-batch size, while our language models have a $2 \times 10^{-4}$ learning rate for the pre-train stage, $10^{-4}$ for the instruction tuning stage, and a 16 mini-batch size for both stages. The motion tokenizer undergoes 150K iterations of training, while the language model undergoes 300K iterations during the pre-train stage and another 300K iterations during the instruction tuning stage. All models are trained on 8 Tesla V100 GPUs.

### 4.2 Comparisons on Motion-relevant Tasks

**Comparisons on Multiple Tasks.** By introducing a uniform framework that treats human motion as a foreign language, we open up the exploration of diverse motion-relevant tasks. We employ a 220M pre-trained *Flan-T5-Base*[38, 5] model as our backbone and fine-tune the model through the pre-training and instruction tuning stage (Sec. 3.3) for all following comparisons. As shown in Tab. 2, we evaluate MotionGPT against state-of-the-art methods on key tasks such as text-conditioned motion generation [54, 59, 12, 48], motion captioning [12], motion prediction [48], and motion in-between[48]. While we leverage existing results from previous works or benchmarks for text-to-motion and motion-to-text tasks, we re-implement the motion diffusion models [48] for motion prediction and evaluate it under the same metrics and settings. Please note that some methods are

| Methods | RPrecision↑ | | | FID↓ | MMDist↓ | Diversity→ | MModality↑ |
|---|---|---|---|---|---|---|---|
| | Top1 | Top2 | Top3 | | | | |
| Real | $0.511^{\pm.003}$ | $0.703^{\pm.003}$ | $0.797^{\pm.002}$ | $0.002^{\pm.000}$ | $2.974^{\pm.008}$ | $9.503^{\pm.065}$ | - |
| TM2T [12] | $0.424^{\pm.003}$ | $0.618^{\pm.003}$ | $0.729^{\pm.002}$ | $1.501^{\pm.017}$ | $3.467^{\pm.011}$ | $8.589^{\pm.076}$ | $2.424^{\pm.093}$ |
| T2M [11] | $0.457^{\pm.002}$ | $0.639^{\pm.003}$ | $0.740^{\pm.003}$ | $1.067^{\pm.002}$ | $3.340^{\pm.008}$ | $9.188^{\pm.002}$ | $2.090^{\pm.083}$ |
| MotionDiffuse [60] | $0.491^{\pm.001}$ | $\mathbf{0.681}^{\pm.001}$ | $\mathbf{0.782}^{\pm.001}$ | $0.630^{\pm.001}$ | $3.113^{\pm.001}$ | $9.410^{\pm.049}$ | $1.553^{\pm.042}$ |
| MDM [48] | $0.320^{\pm.005}$ | $0.498^{\pm.004}$ | $0.611^{\pm.007}$ | $0.544^{\pm.044}$ | $5.566^{\pm.027}$ | $9.559^{\pm.086}$ | $2.799^{\pm.072}$ |
| MLD [54] | $0.481^{\pm.003}$ | $0.673^{\pm.003}$ | $0.772^{\pm.002}$ | $0.473^{\pm.013}$ | $3.196^{\pm.010}$ | $9.724^{\pm.082}$ | $2.413^{\pm.079}$ |
| T2M-GPT [59] | $0.491^{\pm.003}$ | $0.680^{\pm.003}$ | $0.775^{\pm.002}$ | $\mathbf{0.116}^{\pm.004}$ | $3.118^{\pm.011}$ | $9.761^{\pm.081}$ | $1.856^{\pm.011}$ |
| MotionGPT (Pre-trained) | $0.435^{\pm.003}$ | $0.607^{\pm.002}$ | $0.700^{\pm.002}$ | $0.160^{\pm.008}$ | $3.700^{\pm.009}$ | $9.411^{\pm.081}$ | $\mathbf{3.437}^{\pm.091}$ |
| MotionGPT (Fine-tuned) | $\mathbf{0.492}^{\pm.003}$ | $\mathbf{0.681}^{\pm.003}$ | $0.778^{\pm.002}$ | $0.232^{\pm.008}$ | $\mathbf{3.096}^{\pm.008}$ | $\mathbf{9.528}^{\pm.071}$ | $2.008^{\pm.084}$ |

Table 3: Comparison of text-to-motion on HumanML3D [11]. The empty MModality indicates *Real* motion is deterministic. These methods are sorted by FID. *Pre-trained* and *Fine-tuned* indicate uniform motion-language pre-training and specific fine-tuning on this task. ($cf$. Tab. 2 for notations.)

| Methods | RPrecision↑ | | MMDist↓ | $\text{Length}_{\text{avg}}$↑ | Bleu@1↑ | Bleu@4↑ | Rouge↑ | Cider↑ | BertScore↑ |
|---|---|---|---|---|---|---|---|---|---|
| | Top1 | Top3 | | | | | | | |
| Real | 0.523 | 0.828 | 2.901 | 12.75 | - | - | - | - | - |
| TM2T[12] | 0.516 | 0.823 | 2.935 | 10.67 | **48.9** | 7.00 | **38.1** | 16.8 | 32.2 |
| MotionGPT (Ours) | **0.543** | **0.827** | **2.821** | **13.04** | 48.2 | **12.47** | 37.4 | **29.2** | **32.4** |

Table 4: Comparison of motion captioning on HumanML3D [11]. The evaluation metrics follow [12], while we use the ground truth texts without pre-processing for linguistic metrics calculation.

designed for specific tasks, and thus some metrics are empty for tasks they cannot handle. The results presented in Tab. 2 demonstrate that our MotionGPT achieves competitive performance across all evaluated tasks, highlighting its capability to address diverse motion tasks within a single model.

**Comparisons on Text-to-Motion.** The text-to-motion task involves generating human motion sequences based on a given text input. We evaluate the proposed the MotionGPT model as the pre-trained MotionGPT, the same one in Tab. 2, as well as fine-tuned it on text-to-motion task. We compare our MotionGPTs with other SOTAs [12, 11, 48, 54, 59] and evaluate the performance on both HumanML3D and KIT datasets using suggested metrics [11]. The results are computed with a 95% confidence interval, obtained from 20 repeated runs. The majority of the reported results are taken directly from their own papers or the benchmark presented in [11]. Tab. 3 summarizes the comparison results, where MotionGPT achieves competitive performance on most metrics.

**Comparisons on Motion-to-Text.** The motion-to-text task involves generating a text description based on a given human motion sequence. We compare the pre-trained MotionGPT with recent work TM2T [12]. We evaluate the performance on the HumanML3D using the suggested metrics from [12]. Additionally, we measure the average numbers of words $\text{Length}_{\text{avg}}$ for further comparisons. Please note that the reported results in [12] are evaluated with pre-processed ground truth text, which ignores the grammatical tense and plural forms of words. In Tab. 4, we directly use the ground truth text descriptions for a more accurate assessment. This comparison shows that MotionGPT overperforms recent work on text descriptions of given motions.

**Comparisons on Motion Prediction and In-between.** We summarize motion prediction and in-between together as general motion completion. To evaluate the motion completion capability of MotionGPT, we employ part of the AMASS dataset [26], a motion-only dataset. For motion prediction task, we only input around the first 20% of the motion sequence as conditions. For in-between, we mask about 50% motion randomly for completion. We also fine-tune MotionGPT specifically for this task and employ FID, ADE, and FDE as metrics like Sec. 4.1. Furthermore, we evaluate MDM [48] on motion prediction by utilizing their provided model, which also supports motion in-between through masked motion "in-painting". The real motion data is used as one of our baselines. Tab. 5 reports that our MotionGPT has the best motion completion quality and diversity.

| Methods | Motion Prediction | | | | Motion In-between | | |
|---|---|---|---|---|---|---|---|
| | FID ↓ | Diversity↑ | ADE↓ | FDE↓ | FID ↓ | Diversity↑ | ADE↓ |
| Real | 0.002 | 9.503 | - | - | 0.002 | 9.503 | - |
| MDM[48] | 6.031 | 7.813 | 5.446 | 8.561 | 2.698 | 8.420 | 3.787 |
| MotionGPT (Ours) | **0.905** | **8.972** | **4.745** | **6.040** | **0.214** | **9.560** | **3.762** |

Table 5: Comparison of motion prediction and motion in-between on part of AMASSS [26] dataset using motion data only. FID indicates motion quality and Diversity (DIV) for motion diversity within each condition. ADE and FDE are joints distance between generation and ground truth.

| Size | Instruction Tuning | Text-to-Motion | | | Motion-to-Text | | | Motion Prediction | | Motion In-between | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | R TOP3 ↑ | FID ↓ | DIV → | MMDist↓ | Bleu@4↑ | Cider↑ | FID ↓ | DIV → | FID ↓ | DIV → |
| Real | - | 0.797 | 0.002 | 9.503 | 2.901 | - | - | 0.002 | 9.503 | 0.002 | 9.503 |
| Small | | 0.706 | 0.727 | 9.264 | **2.748** | 12.02 | 24.9 | - | - | - | - |
| Small | ✓ | 0.663 | 0.336 | 9.239 | 2.931 | 10.54 | 24.3 | 0.954 | 8.727 | 0.326 | 9.618 |
| Base | | **0.722** | 0.365 | 9.407 | 2.821 | **12.47** | **29.2** | - | - | - | - |
| Base | ✓ | 0.700 | 0.160 | **9.411** | 3.019 | 11.42 | 28.2 | 0.905 | 8.972 | **0.214** | **9.560** |
| Large | | 0.694 | 0.234 | 9.310 | 2.776 | 12.44 | 28.5 | - | - | - | - |
| Large | ✓ | 0.708 | **0.159** | 9.301 | 3.011 | 11.71 | 29.1 | **0.556** | **8.975** | 0.223 | 9.358 |

Table 6: Evaluation of instruction tuning and different model sizes of MotionGPTs in four motion tasks on HumanML3D [11] dataset. ($cf$. Tab. 2 for metrics details)

### 4.3 Ablation Studies

MotionGPT employs T5 [38] as the motion-aware language backbone model, and we train these models with pre-training and then instruction tuning. Thus, both model size and training strategy influence the performance of MotionGPTs. We here evaluate them on the typical motion tasks. More detailed ablation studies are provided in the supplements.

**Model Sizes.** We evaluate the performance of models with different sizes across four motion tasks. Besides the base 220M MotionGPT in Sec. 4.1, we now evaluate 60M, 220M, and 770M MotionGPTs. Tab. 6 demonstrates that the 220M base model has achieved remarkable performance compared to the smaller 60M model. However, the larger model size of current Motions does not yield significant improvements and, in few cases, even leads to worse results, as observed in the motion in-between task. We believe this could be caused by the small amount of current motion datasets. HumanML3D only includes 15k motion sequences, much smaller than even billions of language and image data.

**Effectiveness of Instruction Tuning.** We evaluate the impact of our instruction tuning strategy on different model sizes. The results in Tab. 6 demonstrate that instruction tuning enhances the versatility of MotionGPT, enabling more motion tasks like motion completion and improving the motion performance of the text-to-motion task. However, for pure text-generation tasks, the model performance is downgraded, likely due to the pair amount of textual descriptions and coupled motions.

## 5 Disscusion

As the first trial, to our best knowledge, exploring human motion generation using language models, the proposed MotionGPT still owns limitations as follows. MotionGPT only utilizes motion on articulated human bodies, while many other works focus on faces [17, 4], hands [41, 23, 22] and even animal [42, 64] motion. Besides, our method is also restricted to multiple humans without modeling human-object, or human-environment interactions [43]. It is interesting to model the human interaction scenarios in a motion-language framework and generate controllable motions [43].

We summarize the proposed MotionGPT as a uniform motion-language framework to generate plausible human motion and natural language descriptions through prompt-based instructions. Compared to the compatible motion diffusion methods [54, 48], our MotionGPT produces competitive results on motion generation, motion captioning, motion prediction, and motion in-between using only one pre-trained generative model. With the advancement of large language data and models [38, 5], MotionGPT is also capable of addressing natural question-to-answer tasks. Extensive experiments on various human motion-relevant tasks demonstrate the effectiveness and extendibility of MotionGPT.

## References


[1] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pages 719–728. IEEE, 2019.

[2] Stephen H Bach, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, et al. Promptsource: An integrated development environment and repository for natural language prompts. *arXiv preprint arXiv:2202.01279*, 2022.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[4] Xuan Cao, Zhang Chen, Anpei Chen, Xin Chen, Shiying Li, and Jingyi Yu. Sparse photometric 3d face reconstruction guided by morphable models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4635–4644, 2018.

[5] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022.

[6] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[8] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. *arXiv preprint arXiv:2305.05665*, 2023.

[9] Yusuke Goutsu and Tetsunari Inamura. Linguistic descriptions of human motion with generative adversarial seq2seq learning. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4281–4287. IEEE, 2021.

[10] John C Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

[11] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5152–5161, June 2022.

[12] Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In *ECCV*, 2022.

[13] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020.

[14] Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. Audioclip: Extending clip to image, text and audio. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 976–980. IEEE, 2022.

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[16] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*, 2023.

[17] Tero Karras, Timo Aila, Samuli Laine, Antti Herva, and Jaakko Lehtinen. Audio-driven facial animation by joint end-to-end learning of pose and emotion. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017.

[18] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. Flame: Free-form language-based motion synthesis & editing. *arXiv preprint arXiv:2209.00349*, 2022.

[19] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*, 2018.

[20] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023.

[21] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022.

[22] Yuwei Li, Minye Wu, Yuyao Zhang, Lan Xu, and Jingyi Yu. Piano: A parametric hand bone model from magnetic resonance imaging. *arXiv preprint arXiv:2106.10893*, 2021.

[23] Yuwei Li, Longwen Zhang, Zesong Qiu, Yingwenqi Jiang, Nianyi Li, Yuexin Ma, Yuyao Zhang, Lan Xu, and Jingyi Yu. Nimble: a non-rigid hand model with bones and muscles. *ACM Transactions on Graphics (TOG)*, 41(4):1–16, 2022.

[24] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[25] Hengbo Ma, Jiachen Li, Ramtin Hosseini, Masayoshi Tomizuka, and Chiho Choi. Multi-objective diverse human motion prediction with knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8161–8171, 2022.

[26] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[27] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–8, 2022.

[28] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[29] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[30] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[31] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3D human motion synthesis with transformer VAE. In *International Conference on Computer Vision (ICCV)*, 2021.

[32] Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision (ECCV)*, 2022.

[33] Matthias Plappert, Christian Mandery, and Tamim Asfour. The kit motion-language dataset. *Big Data*, 4(4):236–252, dec 2016.

[34] Matthias Plappert, Christian Mandery, and Tamim Asfour. Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks. *Robotics and Autonomous Systems*, 109:13–26, 2018.

[35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[36] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[37] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[38] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

[39] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and BjÃ¶rn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[41] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022.

[42] Nadine Rueegg, Silvia Zuffi, Konrad Schindler, and Michael J. Black. BARC: Learning to regress 3D dog shape from images by exploiting breed information. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3876–3884, June 2022.

[43] Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H Bermano. Human motion diffusion as a generative prior. *arXiv preprint arXiv:2303.01418*, 2023.

[44] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando: 3d dance generation by actor-critic gpt with choreographic memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11050–11059, 2022.

[45] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[46] Wataru Takano and Yoshihiko Nakamura. Statistical mutual conversion between whole body motion primitives and linguistic sentences for human motions. *The International Journal of Robotics Research*, 34(10):1314–1328, 2015.

[47] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 358–374. Springer, 2022.

[48] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Amit H Bermano, and Daniel Cohen-Or. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022.

[49] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[50] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[51] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.

[52] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022.

[53] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

[54] Chen Xin, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, Jingyi Yu, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023.

[55] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*, 2021.

[56] Tatsuro Yamada, Hiroyuki Matsunaga, and Tetsuya Ogata. Paired recurrent autoencoders for bidirectional translation between robot actions and linguistic descriptions. *IEEE Robotics and Automation Letters*, 3(4):3441–3448, 2018.

[57] Kim Youwang, Kim Ji-Yeon, and Tae-Hyun Oh. Clip-actor: Text-driven recommendation and stylization for animating human meshes. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 173–191. Springer, 2022.

[58] Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 346–364. Springer, 2020.

[59] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. T2m-gpt: Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[60] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022.

[61] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

[62] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

[63] Yan Zhang, Michael J Black, and Siyu Tang. We are more than our joints: Predicting how 3d bodies move. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3372–3382, 2021.

[64] Silvia Zuffi, Angjoo Kanazawa, and Michael J. Black. Lions and tigers and bears: Capturing non-rigid, 3D, articulated shape from images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3955–3963. IEEE Computer Society, 2018.

# Appendix

This appendix provides qualitative comparison results (Sec. A), additional experiments (Sec. B) on the components of MotionGPT models, inference time (Sec. C), statistics on motion vocabulary (Sec. D), evaluations on hyperparameters (Sec. E), user study (Sec. F), a protocol for the uniform evaluation (Sec. G), and more implementation details (Sec. H) of MotionGPT models . Please note evaluations on our training scheme (Sec. B.2), elaborations on the difference of T2M-GPT (Sec. B.4), implementation details of motion completion (Sec. B.5), and more metric definitions (Sec. G).

**Video.** We have provided the supplemental video to illustrate our results. In this video, we show 1) comparisons of text-to-motion, 2) comparisons of motion captioning, and 3) more results on motion prediction and other tasks. We suggest watching this video for dynamic motion results.

**Code** is available in supplements. We provide example code files, which include the process of the training and evaluation of our MotionGPT models, as well as several example results. We can hardly upload our large model files, but all codes, data, and pre-trained models will be fully released.

## A Qualitative Results

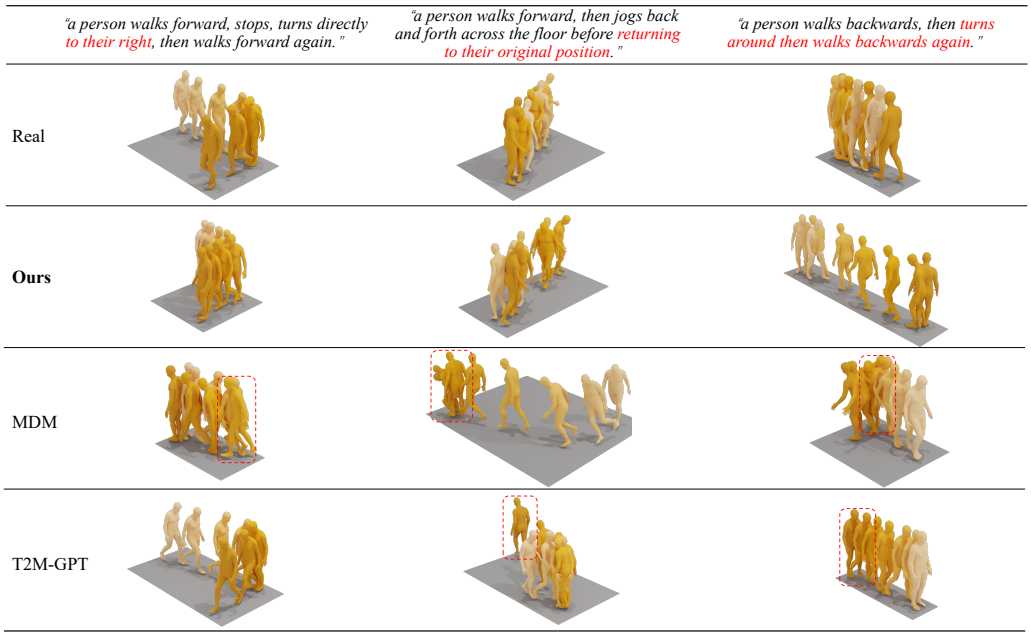



Figure 4: Comparison on text-driven motion generation. The provided state-of-the-art methods are under the same training and inference setting on HumanML3D [11]. The red words and boxes highlight the misaligned motions. The results demonstrate that our motion-language per-training shows promising text understanding for motion generation.

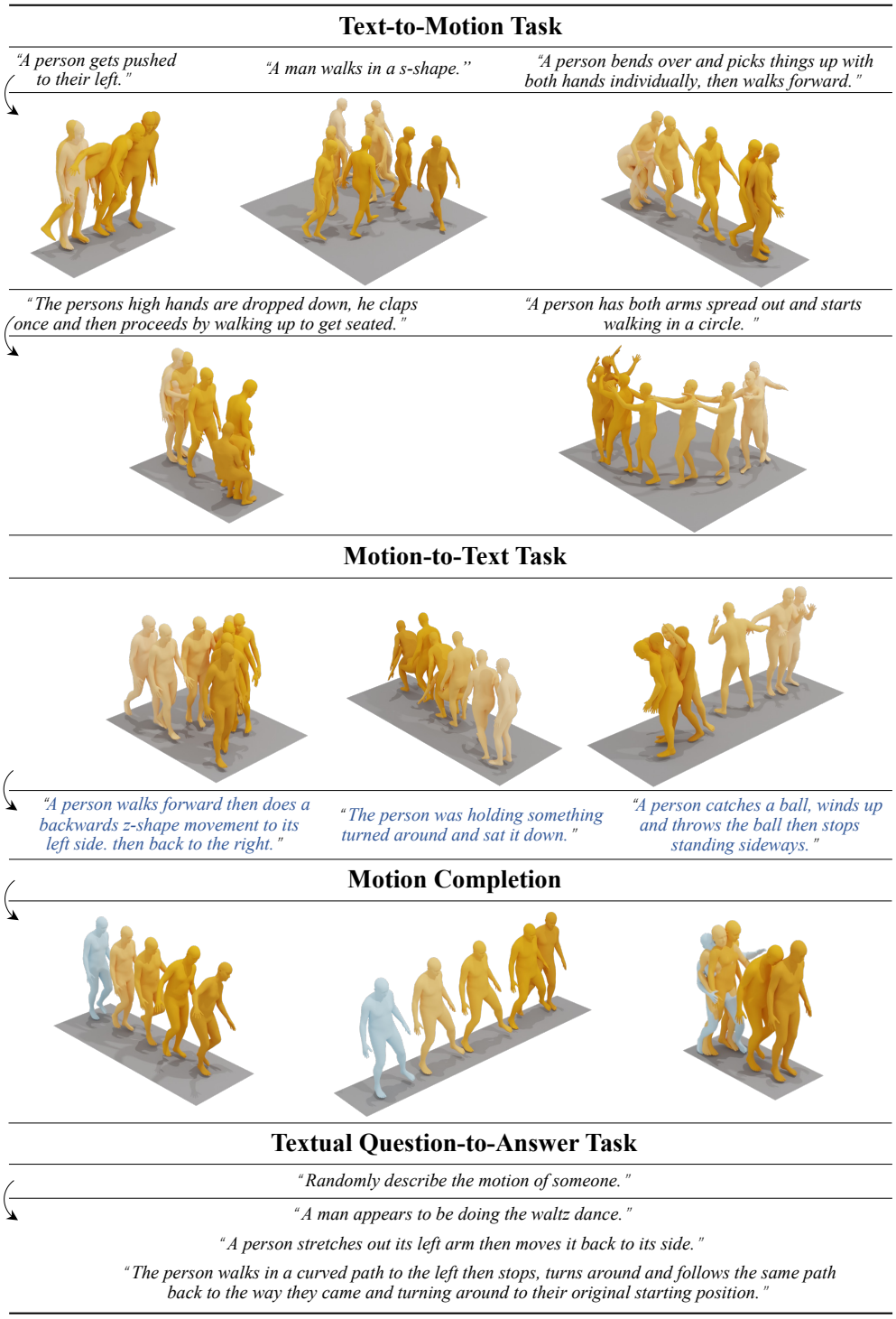



Figure 5: Gallery for the results of our unified MotionGPT. More samples are from our best model for text-to-motion synthesis, motion captioning, and textual question-to-answer task. The supervision of MotionGPT relies on our instruction-based motion-language dataset (*c.f.* Sec. G) based on previous motion datasets [11, 26]. We recommend the dynamic visualization in our supplemental video.

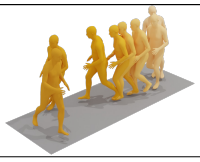
| | | | |
|---|---|---|---|
| Input Motions |  |  |  |
| Real | *"a person walks backwards, then turns around then walks backwards again."* | *"a man starts to walk straight then walks **to the right**."* | *"person is acting like a grizzly **bear**."* |
| TM2T | *"a person walk forward then turn around and walk back"* | *"a person walk in a curved line"* | *"a person raise their arm to their chest multiple time"* |
| **Ours** | *"a person takes two steps forward then turns **to their right** 180 degrees and takes two steps away."* | *"a person walks in a **semi-circular** pattern, **tip-toeing**."* | *"a person pretends to be a **bear**."* |

Figure 6: Comparison of the state-of-the-art method on motion captioning task. All provided methods are under the same training and inference setting on HumanML3D [11]. The results demonstrate that our text descriptions correspond better to the motion and have correct grammar. The orange words indicate the matching results, while the red marks the incorrect grammar.

# B Additional Experiments

We conduct several experiments to continue the evaluations of MotionGPT models. We first evaluate the text-to-motion results on KIT dataset (Sec. B.1). Then we evaluate the hyperparameters of motion tokenizer $\mathcal{V}$ (Sec. B.2). After that, we study the effectiveness of the training scheme (Sec. B.2). We also provide the elaboration on the difference of T2M-GPT (Sec. B.4), implementation details of motion completion (Sec. B.5).

## B.1 Text-to-Motion on KIT dataset.

Following the same procedure on HumanML3D[11] dataset, We train a 220M MotionGPT base model on the KIT[33] dataset without any pre-training. We evaluate this model under the same settings of [11]. Most results are borrowed from their own paper of the benchmark in [11]. Tab. 7 shows that MotionGPT achieves comparable performance compared to the previous state-of-the-arts.

| Methods | RPrecision↑ | | | FID↓ | MMDist↓ | Diversity→ | MModality↑ |
|---|---|---|---|---|---|---|---|
| | Top1 | Top2 | Top3 | | | | |
| Real | $0.424^{\pm.005}$ | $0.649^{\pm.006}$ | $0.779^{\pm.006}$ | $0.031^{\pm.004}$ | $2.788^{\pm.012}$ | $11.08^{\pm.097}$ | - |
| TM2T[12] | $0.280^{\pm.005}$ | $0.463^{\pm.006}$ | $0.587^{\pm.005}$ | $3.599^{\pm.153}$ | $4.591^{\pm.026}$ | $9.473^{\pm.117}$ | $3.292^{\pm.081}$ |
| MDM[48] | $0.164^{\pm.004}$ | $0.291^{\pm.004}$ | $0.396^{\pm.004}$ | $0.497^{\pm.021}$ | $9.191^{\pm.022}$ | $10.85^{\pm.109}$ | $1.907^{\pm.214}$ |
| MLD[54] | $0.390^{\pm.008}$ | $0.609^{\pm.008}$ | $0.734^{\pm.007}$ | $0.404^{\pm.027}$ | $3.204^{\pm.027}$ | $10.80^{\pm.117}$ | $2.192^{\pm.071}$ |
| T2M-GPT [59] | $0.416^{\pm.006}$ | $0.627^{\pm.006}$ | $0.745^{\pm.006}$ | $0.514^{\pm.029}$ | $3.007^{\pm.023}$ | $10.92^{\pm.108}$ | $1.570^{\pm.039}$ |
| MotionGPT (Ours) | $0.366^{\pm.005}$ | $0.558^{\pm.004}$ | $0.680^{\pm.005}$ | $0.510^{\pm.016}$ | $3.527^{\pm.021}$ | $10.35^{\pm.084}$ | $2.328^{\pm.117}$ |

Table 7: We involve KIT [33]dataset and evaluate the methods on the text-driven motion generation task. Please refer to Tab. 3 for more details on metrics and notations.

## B.2 Ablation on Motion Tokenizer.

We ablate the motion tokenizer $\mathcal{V}$ of our MotionGPT models, studying the size $K$ of motion codebooks. We also compare this VQ-VAE with other VAE models in previous works [30, 31, 54], as shown in Tab. 8. This comparison demonstrates the improvement of VQ-VAE on motion reconstruction. With this ablation studies on the codebook size $K$, we thus select $K = 512$ for most experiments.

| Method | Reconstruction | | | | |
|---|---|---|---|---|---|
| | MPJPE↓ | PAMPJPE↓ | ACCL↓ | FID↓ | DIV→ |
| Real | - | - | - | 0.002 | 9.503 |
| VPoser-t [30] | 75.6 | 48.6 | 9.3 | 1.430 | 8.336 |
| ACTOR [31] | 65.3 | 41.0 | **7.0** | 0.341 | **9.569** |
| MLD-1 [54] | **54.4** | 41.6 | 8.3 | 0.247 | 9.630 |
| MotionGPT (Ours) | 55.8 | **40.1** | 7.5 | **0.067** | 9.675 |
| $K = 256$ | 76.4 | 51.3 | 10.0 | 0.187 | **9.496** |
| $K = 512$ | **55.8** | **40.1** | **7.5** | **0.067** | 9.675 |
| $K = 1024$ | 60.3 | 44.0 | 8.6 | 0.086 | 9.677 |
| $K = 2048$ | 78.9 | 51.4 | 10.5 | 0.141 | 9.597 |

Table 8: Evaluation of our motion tokenizer on the motion part of HumanML3D [11] dataset. We follow MLD [54] to evaluate our VQ-VAE model $\mathcal{V}$: MPJPE and PAMPJPE are measured in millimeter. ACCL indicates acceleration error. We evaluate FID and Diversity the same as Tab. 3. The baselines of VPoser-t [30] and ACTOR [31] are borrowed from MLD. $K$ indicates the codebook size, and $K = 512$ shows the best performance of motion reconstruction.

### B.3 Effectiveness of Training Scheme

**Motion-Language Pre-training** vs **Instructions Tuning**. We have provided the illustration of our training scheme in Fig. 3 and the evaluation in Tab. 6. We further ablate this training scheme on the base MotionGPT model, by evaluating the motion-language pre-training (the second step) and instruction tuning (the third step). As shown in Tab. 9, we train these models with the same 600K iterations. Compared to other training combinations, the full-stage MotionGPT achieves higher performance on most motion tasks.

| Size | Pre-training | Instruction Tuning | Text-to-Motion | | | Motion-to-Text | | | Motion Prediction | | Motion In-between | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | R TOP3 ↑ | FID ↓ | DIV → | MMDist↓ | Bleu@4↑ | Cider↑ | FID ↓ | DIV → | FID ↓ | DIV → |
| Real | - | - | 0.797 | 0.002 | 9.503 | 2.901 | - | - | 0.002 | 9.503 | 0.002 | 9.503 |
| Base | ✔ | ✗ | **0.722** | 0.365 | 9.407 | **2.821** | **12.47** | **29.2** | - | - | - | - |
| Base | ✔ | ✔ | 0.700 | **0.160** | **9.411** | 3.019 | 11.42 | 28.2 | 0.905 | 8.972 | **0.214** | **9.560** |
| Base | ✗ | ✔ | 0.607 | 0.324 | 9.563 | 3.374 | 10.92 | 27.7 | 1.643 | 8.829 | 0.323 | 9.628 |

Table 9: Evaluation of the training scheme on the base MotionGPT models. We evaluate the results with the proposed evaluation protocols in Sec. G. Please refer to Tab. 2 for metrics and the details.

**Instructions Tuning** vs **Task-Specific Tuning**. While our unified instruction-tuned MotionGPT model has demonstrated competitive performance across various motion-related tasks, further fine-tuning can always enhance its performance on specific tasks. Therefore, we focus on the text-to-motion task and motion in-between task as illustrative examples to showcase the performance of the model before and after fine-tuning. By comparing the results in Tab. 10, we can assess the effectiveness of fine-tuning in improving task-specific performance.

| Insturct tuned | Fine tuned | Text-to-Motion | | | Motion In-between | | |
|---|---|---|---|---|---|---|---|
| | | R TOP1↑ | FID↓ | DIV→ | FID ↓ | DIV↑ | ADE↓ |
| ✔ | ✗ | 0.435 | **0.160** | 9.411 | 0.214 | **9.560** | 3.762 |
| ✔ | ✔ | **0.492** | 0.232 | **9.528** | **0.209** | 9.378 | **3.281** |

Table 10: Evaluation of new task tuning of different size models on HumanML3D [11] dataset.

### B.4 Difference of T2M-GPT

We introduce the difference between T2M-GPT [59] to show our unified framework. T2M-GPT investigates a generative framework based on VQ-VAE and Transformer for motion generation only. They incorporate language information by leveraging CLIP [35] to extract text embedding as motion generation conditions, which is similar to most previous work, such as MDM [48], MLD [54], and MotionDiffuse [60]. However, our MotionGPTs are based on the pre-trained language model so it naturally leverages the strong language generation and zero-shot transfer abilities of pre-trained language models. Benefiting from the motion-language vocabulary, MotionGPT thus generates both human language and human motion in a unified model.

### B.5 Implementation details of Motion Completion

Please note that MDM[48] accomplish motion in-between task in their paper through masked motion "in-painting" which fix the first and last 25% of the motion, leaving the model to generate the remaining 50% in the middle. To achieve the motion prediction task with MDM, we fix the first 20% of the motion and then generate the remaining. All our results are computed by utilizing their provided pre-trained model. To compare with MDM in Tab. 5 on both motion in-between and motion prediction tasks, we evaluate our MotionGPT with the same setting during the inference.

## C Inference Time

We provide a detailed study on inference time with our different model sizes below. Due to our auto-regressive model for motion generation, we use Frames Per Second (FPS) to evaluate our time costs. All the time costs are evaluated on 8 Tesla V100 using one batch size. Tab. 11 shows that any size of our MotionGPTs can support real-time human animations and come up to hundreds of FPS.

| Models | Backbone | Parameters | FPS ↑ |
|---|---|---|---|
| MotionGPT | Small | 60 M | 421.31 |
| MotionGPT | Base | 220 M | 222.69 |
| MotionGPT | Large | 770 M | 119.75 |

Table 11: Evaluation of inference time costs on text-driven motion generation. We evaluate the Frames Per Second (FPS) by averaging our generated frames for each second. We show the time costs on different model sizes. Under the same 1 Tesla V100, the smaller model size gets the faster FPS. All models can support real-time motion animation applications.

## D Statistics on Motion Vocabulary

We visualize the usage of each "word" in our motion vocabulary $V_m$ item generated by our motion tokenizer $\mathcal{V}$. We sample all motions from the whole test set of HumanML3D dataset [11] and count each "word". In Fig. 7, it shows the utilization of our motion codebook, which seems to be a concise but informative motion vocabulary.

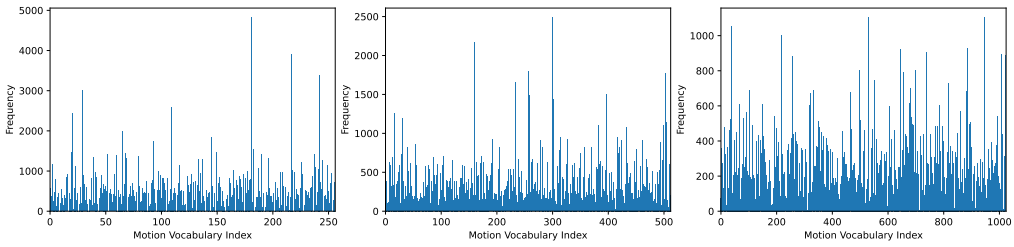



Figure 7: The statistics of each "word" in different sizes of motion vocabulary $V_m$. From left to right, the vocabulary size is $K = 256, 512, 1024$. ($cf$. Tab. 8, $K = 512$ for the best motion quality.)

## E Evaluation of Hyperparameters

We conduct experiments to investigate the impact of different sampling strategies on the generation results. Specifically, we compare the use of greedy search, which selects the most probable token at each step, with sampling from the probability distribution and adopting beam search, which is evaluated in previous language models [38]. Beam search expands the search space for improved sequence probability matching. The results in Tab. 12 demonstrate that while avoiding sampling and using beam search can slightly improve generation quality, they also significantly reduce the diversity of generated motions from the same text description.

| Method | Sample | #beams | R Precision Top 3↑ | FID↓ | MM Dist↓ | Diversity→ | MModality↑ |
|---|---|---|---|---|---|---|---|
| Real | - | - | $0.797^{\pm.002}$ | $0.002^{\pm.000}$ | $2.974^{\pm.008}$ | $9.503^{\pm.065}$ | - |
| MotionGPT | | - | $0.780^{\pm.002}$ | $0.224^{\pm.009}$ | $3.076^{\pm.009}$ | $9.492^{\pm.056}$ | - |
| | | 2 | $0.780^{\pm.002}$ | $0.199^{\pm.008}$ | $3.083^{\pm.007}$ | $9.512^{\pm.063}$ | - |
| | | 3 | $0.781^{\pm.002}$ | $0.179^{\pm.008}$ | $3.099^{\pm.009}$ | $9.516^{\pm.064}$ | - |
| | | 4 | $0.782^{\pm.002}$ | $0.160^{\pm.007}$ | $3.092^{\pm.010}$ | $9.536^{\pm.060}$ | - |
| MotionGPT | ✓ | - | $0.778^{\pm.002}$ | $0.232^{\pm.008}$ | $3.096^{\pm.008}$ | $9.528^{\pm.071}$ | $2.008^{\pm.084}$ |
| | ✓ | 2 | $0.780^{\pm.002}$ | $0.194^{\pm.008}$ | $3.091^{\pm.010}$ | $9.508^{\pm.063}$ | $1.140^{\pm.064}$ |
| | ✓ | 3 | $0.780^{\pm.002}$ | $0.190^{\pm.008}$ | $3.089^{\pm.011}$ | $9.529^{\pm.061}$ | $0.929^{\pm.055}$ |
| | ✓ | 4 | $0.780^{\pm.002}$ | $0.182^{\pm.008}$ | $3.093^{\pm.008}$ | $9.537^{\pm.059}$ | $0.803^{\pm.044}$ |

Table 12: Evaluations on hyperparameters for MotionGPT generations. We study the influence of two hyperparameters: *sample* stands for sampling from distribution; *#beams* means the number of beams for beam search, where empty means no beam search.

## F User Study

For the comparisons of text-to-motion task, we use the force-choice paradigm to ask "Which of the two motions is more realistic?" and "which of the two motions corresponds better to the text prompt?". The provided motions are generated from 30 text descriptions from the test set of HumanML3D [11] dataset. For the comparisons of motion-to-text task, we ask 15 users to choose the motion descriptions from GT, TM2T [12], and our MotionGPT. The motions are from the test set of HumanML3D [11] dataset. As shown in Fig. 8, in both two tasks, our MotionGPT was preferred over the other state-of-the-art methods and even competitive with the ground truth.

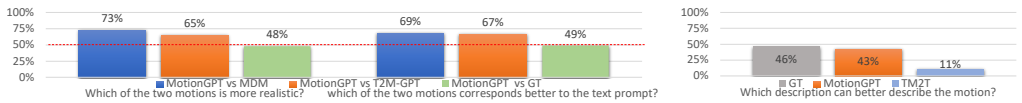



Figure 8: User Study. We investigate our motion quality and the alignment with test descriptions. The left part is the user study for text-to-motion. The right part is for motion captioning.

## G Evaluation Protocols on the Uniform Motion-Language Generation.

We propose a protocol to evaluate our unified MotionGPT on multiple motion-language generation tasks. Upon previous datasets [11, 33, 26], we build an instruction motion-language dataset, which is composed of 14 core tasks (Fig. 9) for now. As shown in Tab. 13, each core task has dozens of instruction prompts (Tab. 13). We will release the pre-processed dataset.

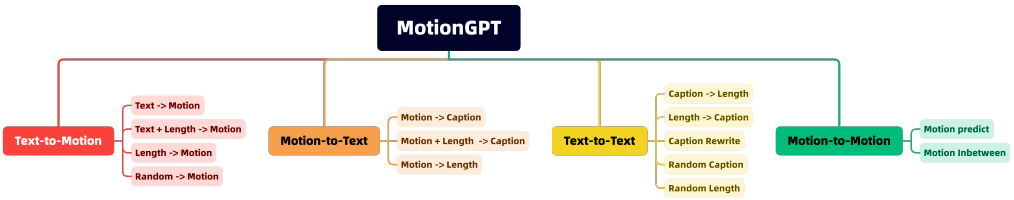



Figure 9: Protocols for multiple motion-language tasks. For each task, we follow Tab. 13 to process the previous datasets [26, 11] into the instruction-based data.

| Task | Input | Output |
|---|---|---|
| Text-to-Motion | Give me a motion that corresponds to [caption].<br>Demonstrate a sequence of movements that depict [caption].<br>I need a human motion that conveys [caption]. Can you generate it for me? | [motion] |
| Text-to-Motion w/ length | Give me a motion that lasts for approximately [frames] frames. The caption is: [caption].<br>Please create a motion that lasts [seconds] seconds and illustrates [caption]. | [motion] |
| Length-to-Motion | Show me a motion that lasts for no more than [frames] frames.<br>Create a motion that has a duration of [seconds] seconds. | [motion] |
| Radnom Motion | Give me motions as you like.<br>Produce actions that are not prescribed. | [motion] |
| Motion-to-Text | Give me a summary of the motion being displayed in [motion] using words.<br>Describe the motion illustrated in [motion] in natural language. | [caption] |
| Motion-to-Text w/ length | Describe the movement portrayed in [motion]that lasts [frames] frames.<br>What is happening in [motion] for a length of [seconds] seconds? | [caption] |
| Motion-to-Length | What is the duration of [motion]'s gestures in frames?<br>What is the total duration of [motion]'s body movements in seconds? | There are [frames] frames in the motion.<br>The motion lasts for [seconds] seconds. |
| Caption-to-Length | How many frames are expected for the motion that matches [caption]?<br>Given [caption], provide the anticipated second duration for the corresponding motion. | The duration is estimated to be around [frames] frames.<br>The motion has a length of [seconds] seconds. |
| Length-to-Caption | What are some possible physical gestures that could be made in [frames] frames?<br>What motion could be performed in [seconds] seconds? | [caption] |
| Random Caption | Depict a motion as like you have seen it.<br>Describe the motion of someone randomly. | [caption] |

Table 13: Some examples of prompt templates in our uniform evaluation protocols.

**Metric Definitions:** We provide more details of evaluation metrics as follows. Our evaluation metrics can roughly divide to five classes including text-motion matching, generation diversity, linguistic quality, motion quality, and time cost. For the first two classes, [54] has already claims clearly and for the linguistic metrics including BLUE [29], Rouge [24], Cider [51], and BertScore [62], you can refer to their own papers for details. Here we focus on the explaination of the rest metrics.

**Motion Quality**. FID, MPJPE, PAMPJPE [10], ACCL have been clearly explained in [54]. Thus here we focus on the Average Displacement Error (ADE) and Final Displacement Error (FDE) refaccuracy of the predicted motion. Following previous motion prediction work[58, 63, 25], ADE is defined as average L2 distance between the ground truth and predicted motion of the whole sequence and FDE is the L2 distance between the ground truth and predicted motion in the last frame.

**Time Costs**. To evaluate the computing efficiency of our models, especially the inference efficiency, we calculate average Frames Per Second (FPS) when generating motions. In our case, we calculate FPS on the test set of HumanML3D [11], set the batch size to one, and ignore the time cost for model and dataset loading parts.

# H Details on MotionGPT Models

## H.1 Implementation Details

Besides the MotionGPt with 220M parameters, we implement a smaller model that reduces the model dimension with $d_{\text{model}} = 512$, $d_{\text{ff}} = 2048$ with only 6 layers in encoder and decoder, as well as a larger model with 770 million parameters, which increases the model dimensions with $d_{\text{model}} = 1024$, $d_{\text{ff}} = 4096$, $d_{\text{kv}} = 64$, 24 layers for each transformer. Except for the training iterations during the instruction tuning stage, the other settings are the same. Please refer to Tab. 14 for more details.

| MotionGPT | Small | Base | Large |
|---|---|---|---|
| Backbone | Flan-T5-Small | Flan-T5-Base | Flan-T5-Large |
| Training Batch Size | 64 | 16 | 4 |
| Model Size | 60M | 220M | 770M |
| Pre-training - Iterations | 300K | 300K | 300K |
| Pre-training - Learning Rate | 2e-4 | 2e-4 | 2e-4 |
| Instruction Tuning - Iterations | 200K | 300K | 400K |
| Instruction Tuning - Learning Rate | 1e-4 | 1e-4 | 1e-4 |
| Motion Vocabulary Number $V_m$ | 512 | 512 | 512 |
| Motion Codebook Dimension | 512 | 512 | 512 |

Table 14: Hyperparameters for different MotionGPTs. We train these models on 64 Tesla V100 GPUs. The smaller model size lowers the computational requirements and thus provides faster inference ($cf.$ Sec. C). According to Tab. 6, the base MotionGPT model is the best one for overall tasks. However, we believe this could be caused by the small amount of current motion datasets. The large model could achieve the best performance when the amount of data comes up to millions or even billions.