CLaM: An Open-Source Library for Performance Evaluation of Text-driven Human Motion Generation

Xiaodong Chen[†]
University of Science and Technology of China
Hefei, China
cxd1230@mail.ustc.edu.cn

Kunlang He[†]
JD Explore Academy
Beijing, China
allenhethis@outlook.com

Wu Liu[⊠]
University of Science and Technology of China
Hefei, China
liuwu@live.cn

Xinchen Liu
JD Explore Academy
Beijing, China
liuxinchen1@jd.com

Zheng-Jun Zha
University of Science and Technology of China
Hefei, China
zhazj@ustc.edu.cn

Tao Mei
HiDream.ai Inc.
Beijing, China
tmei@live.com

ABSTRACT

Text-driven human motion generation, which creates motion sequences based on textual descriptions, has attracted great attention in the communities of multimedia and artificial intelligence. By parsing and comprehending textual information and converting it into specific human movements, it realizes a direct transformation from human semantics to motion sequences. New text-driven human motion generators are springing up to achieve better performance. However, the absence of well-trained evaluators that can effectively estimate the consistency between the text prompts and motions generated by existing generators remains a challenge. To address the above issues, we propose an open-source library with a powerful Contrastive Language-and-Motion (**CLaM**) pre-training evaluator, which can be employed for evaluating a variety of text-driven human motion generation algorithms. We perform a thorough performance evaluation of the existing algorithms on various metrics, such as R-Precision. As a by-product, we build a large-scale **HumanML3D-synthesis** dataset, which consists of 14,616 motion sequences and 547,102 textual descriptions, which is ten times larger than the widely-used HumanML3D dataset. The source codes and models for CLaM are available at https://github.com/SheldongChen/CLaM/.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**.

KEYWORDS

Contrastive Language-Motion Pre-training, Text-driven Motion Generation, Open-Source Library

[†] Equal contribution. [⊠] Corresponding author.

MM '24, October 28–November 1, 2024, Melbourne, VIC, Australia.
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0686-8/24/10
https://doi.org/10.1145/3664647.3685523

ACM Reference Format:
Xiaodong Chen[†], Kunlang He[†], Wu Liu[⊠], Xinchen Liu, Zheng-Jun Zha, and Tao Mei. 2024. CLaM: An Open-Source Library for Performance Evaluation of Text-driven Human Motion Generation. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM '24), October 28–November 1, 2024, Melbourne, VIC, AustraliaProceedings of the 32nd ACM International Conference on Multimedia (MM'24), October 28-November 1, 2024, Melbourne, Australia.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3664647.3685523

1 INTRODUCTION

Text-driven human motion generation [2, 5, 6, 11] aims to generate human motion sequences that align with the given text descriptions. The innovation and practicality of this technology lie in its integration of language understanding and motion generation. This task has broad applications, including but not limited to digital humans, animation-making, and so on.

Generating motion from textual descriptions is challenging, as motion and text are from different modalities, and a single motion can be described through countless similar sentences. The research on generators focuses on generating more realistic human motion from texts. For example, T2M [2] and TM2T [3] solve the understanding of long sentences and generation of variable-length motions. Motion Diffuse [13] and MLD [1] introduced diffusion models to this task, resulting in improved quality of generated motion sequences. T2M-GPT [12] greatly enhances the semantic comprehension capabilities of the generators by incorporating Generative Pre-trained Transformers (GPT).

Although the quality of generated motion sequences is improving, this field still encounters some unavoidable challenges that must be faced, especially in research on evaluators. Research on evaluators mainly concerns how to more accurately evaluate the alignment between the generated motion and the given description, which is not directly involved in the training of generators, but determines the selection and evaluation of generators. So far, most previous research uses the evaluator provided by Guo et al. [2] for evaluating the alignment between the real motion and corresponding description. However, the evaluator with poor judgment has significantly hindered the development of this field. As shown in Fig. 1, the recall precision (R-Precision) of the default evaluator [2] is only 51.1% between the Ground Truth (GT) motion and

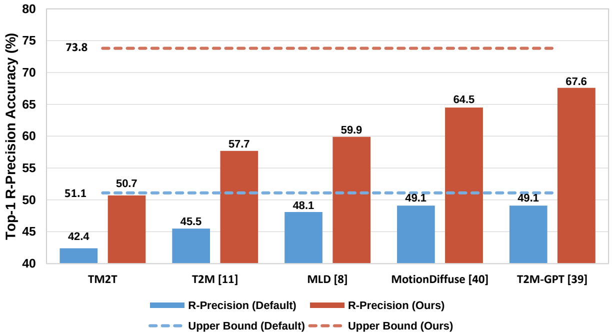

Figure 1: Evaluation results of latest motion generators and upper bound (groundtruth text-motion pairs) with the default evaluator [2] and our CLaM evaluator.

corresponding annotated textual descriptions. This has made it increasingly challenging to assess the latest models (e.g. Motion Diffuse, T2M-GPT) using the prior evaluator fairly, and researchers can only rely on manual qualitative assessment to further compare the generative results of each generator.

To overcome these challenges, we propose an open-source library with a powerful Contrastive Language-and-Motion (**CLaM**) pre-training evaluator, which can perform a thorough performance evaluation of the existing algorithms on various metrics, such as R-Precision. Specifically, our CLaM comprises a text extractor and a motion extractor as the stronger backbone. Additionally, we introduce the plug-and-play synonym provider and auxiliary contrastive loss for the training stage. As shown in Fig. 1, compared to the default evaluator [2] that can hardly distinguish the performance of latest powerful generators [12, 13], it has approximately 22.7% R-Precrion improvement over the default evaluator [2], to ensure that improvements in generators are accurately reflected and substantiated by the reliable evaluator. As a by-product of our plug-and-play LLM-based synonym provider, we launch the HumanML3D-synthesis dataset, which consists of 14,616 motion sequences and 547,102 corresponding textual descriptions, 10 times larger than the HumanML3D dataset.

2 ARCHITECTURE OF CLAM

This section declares the detailed framework of our **CLaM** evaluator. Before describing our method in detail, we first introduce the necessary notations and definitions of the text-driven human motion generation task.

2.1 Preliminary

The task of text-driven human motion generation is, given a textual description X, to generate the corresponding human motion sequences M'. After generation, the evaluator is applied for the performance evaluation. Specifically, the generated motion sequence M' and the given textual description X are processed through the evaluator to extract the motion features $f_{M'}$ and text features f_t, respectively, and to compute metrics such as R-Precision. Following the criteria proposed by Guo et al. [2], the motion feature extractor and text feature extractor of the evaluator is trained under contrastive loss, with the real motion sequence M and corresponding textual description X, to produce geometrically close feature vectors for matched text-motion pairs.

2.2 Contrastive Pre-Training Evaluator CLaM

This subsection introduces our evaluator for text-driven human motion generation named the Contrastive Language-and-Motion pre-training model (CLaM). The design of CLaM is based on the criteria proposed by Guo et al. [2], and its main purpose is to be applied to quantitatively measure the quality of generated motion. As shown in Fig. 2, our CLaM contains a text extractor and a motion extractor, where the text extractor consists of a plug-and-play LLM-based synonym provider, a pretraining tokenizer and a text encoder, and the motion extractor consists of a pretraining conv encoder and a motion encoder.

Synonym Provider. This module expands the original textual descriptions using carefully crafted prompts, thereby enhancing the diversity of text sentences in motion descriptions. It involves three steps: Firstly, constructing appropriate prompts; Secondly, utilizing existing large language model, such as ChatGPT, to generate more enriched motion descriptions; and finally, selecting suitable descriptions for the training process. The essence of text enhancement is synonym sentence building, which is an easy task for ChatGPT. However, for the text-driven human motion generation task, we need to ensure both the accuracy and diversity of our synonymous descriptions. After several trials and manual evaluations, we selected the following prompt.

I would like you to act as a tautology sentence provider. I will tell you several sentences that describe the same human motion and you will give me a list of 20 synonymous sentences based on my prompts. You will only reply to the sentences list (1. ... 2. ... 3. ...), and nothing else. Do not write explanations: [X_1, X_2, X_3], where X_1, X_2, X_3 are different descriptions of the same human motion sequence. Specifically, *'tautology sentence provider'* is added in the prompt to articulate our requirements. *'describe the same human motion'* indicates that X_1, X_2, and X_3 describe the same human motion, which helps the language model to understand the provided motion sequence comprehensively. *'Do not write explanations'* effectively reduces the redundancy of responses.

Text Tokenizer. For the input textual description, we encode it as the token embedding E_t with a pre-training text tokenizer. The tokenizer used in our CLaM is based on the Byte Pair Encoding (BPE) algorithm [10], which is a subword tokenization method. In the encoding phase, the pre-training tokenizer splits the input text into the longest subwords that exist in the pre-training vocabulary as much as possible and then encodes them as token embedding E_t.

Text Encoder. This module is designed for extracting the global text feature f_t from the token embedding E_t. We use the causal self-attention [9] module as the core of our text encoder. Causal self-attention is a variant of the self-attention mechanism that is designed to prevent future information leakage when making predictions. It is often used in tasks like language modeling, which require generating a sequence where the prediction at each step should only depend on previous steps. The core formula of causal self-attention is as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\dot{mask}\right)V\ , \qquad (1)$$

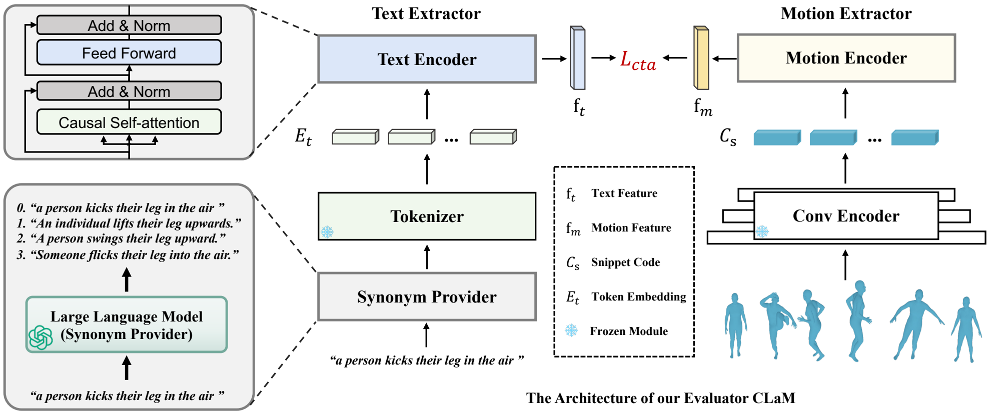
Add & Norm
Feed Forward
Add & Norm
Causal Self-attention
0. "a person kicks their leg in the air "
1. "An individual lifts their leg upwards."
2. "A person swings their leg upward."
3. "Someone flicks their leg into the air."
Large Language Model (Synonym Provider)
"a person kicks their leg in the air "
Text Extractor
Text Encoder
E_t
Tokenizer
Synonym Provider
"a person kicks their leg in the air "
f_t
L_{cta}
f_m
f_t Text Feature
f_m Motion Feature
C_s Snippet Code
E_t Token Embedding
Frozen Module
Motion Extractor
Motion Encoder
C_s
Conv Encoder
The Architecture of our Evaluator CLaM

Figure 2: The architecture of our evaluator CLaM. Our CLaM contains a text extractor and a motion extractor, where the text extractor consists of an LLM-based synonym provider, a pretraining tokenizer and a text encoder, and the motion extractor consists of a pretraining conv encoder and a motion encoder. The conv encoder is pretrained through motion autoencoder [2].

Methods	Year	Type	R-Precision (%) ↑			FID →	MM-Dist →	Diversity →
			Top-1	Top-2	Top-3			
TEMOS [6]	ECCV'22	Retriever	$40.5^{\pm0.3}$	$53.5^{\pm0.3}$	$61.1^{\pm0.2}$	-	-	-
TMR [7]	ICCV'23		$67.2^{\pm0.2}$	$81.3^{\pm0.1}$	$86.8^{\pm0.1}$	-	-	-
Guo et al. [2]	CVPR'22	Evaluator	$51.1^{\pm0.3}$	$70.3^{\pm0.3}$	$79.7^{\pm0.2}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$
CLaM (Ours)	-		**73.8 (+22.7)**$^{\pm0.2}$	**87.0 (+16.7)**$^{\pm0.2}$	**91.7 (+22.0)**$^{\pm0.1}$	**0.004**$^{\pm.000}$	**4.154**$^{\pm.006}$	**7.913**$^{\pm.026}$

Table 1: Comparison with different evaluators and retrievers on HumanML3D [2] test set using ground-truth motion sequences. The evaluation is repeated 20 times, and the mean value is reported, supplemented by a 95% confidence interval. Note that metrics on ground-truth motion sequences are not comparable, except for R-Precision.

where Q, K, and V are the query, key, and value matrices respectively, d_k is the dimension of the keys, and *mask* is a mask matrix that ensures that we only attend to past and current positions. Besides, causal self-attention can be more efficient in tasks that involve sequential data as it does not need to attend to future positions, which can save computation.

Conv Encoder. We encode the motion sequence $M = (m_1, m_2, ..., m_T)$ as snippet code $C_s = (c_s^1, c_s^2, ..., c_s^t)$ through a pre-training conv encoder, which encodes motion sequences in the timeline with 1-D convolution. The conv encoder is trained through the motion autoencoder [2]. In contrast to motion sequence, snippet code C_s has a unified dimension and capsule temporal semantic information, an essential element for the extraction of global motion features.

Motion Encoder. This module is designed to extract the global motion feature, denoted as f_m, from the snippet code C_s. Given the similarity between motion sequences and text sentences, we incorporate causal self-attention into our motion encoder. This approach is nearly identical to that used in the text encoder. Incorporating causal self-attention not only enhances the efficiency of our motion encoder but also enables it to capture the temporal dependencies in the motion sequences more effectively.

Contrastive Training. The loss function for our CLaM model aims to minimize the distance between matched text-motion feature pairs while ensuring that mismatched text-motion feature pairs are dispersed with a margin of at least d_{min}. This goal is achieved through the application of a contrastive loss function. It can be formulated as follows:

$$L_{Con} = y \cdot \|\mathbf{f_m} - \mathbf{f_t}\|_2{}^2 + (1-y) \cdot max(0, d_{min} - \|\mathbf{f_m} - \mathbf{f_t}\|_2)^2 , \quad (2)$$

where y is a binary label equals 1 if the pair is matched, and 0 if it is not. While L_{Con} can constrain the feature space to some extent, it is less efficient in utilizing negative sample data. To overcome this limitation, we introduce InfoNCE as an auxiliary loss function as follows:

$$\mathcal{L}_{\text{NCE}} = -\frac{1}{2N}\sum_i \left(\log\frac{\exp S_{ii}/\tau}{\sum_j \exp S_{ij}/\tau} + \log\frac{\exp S_{ii}/\tau}{\sum_j \exp S_{ji}/\tau}\right) , \quad (3)$$

where N is the number of batch samples, S_{ij} is the cosine distance between samples, and τ is the temperature hyperparameter. The total loss used to train our CLaM is $L_{cta} = \lambda L_{Con} + (1-\lambda)L_{NCE}$ where λ controls the weight of each loss.

3 BENCHMARKING RESULTS AND ANALYSIS

We first introduce benchmark text-to-motion datasets, evaluation metrics in section 3.1. After that, we compare the quantitative results of our CLaM in section 3.2. At last, we provide statistical information on our HumanML3D-synthesis dataset in section 3.3.

3.1 Experimental Setup

Our experiments are conducted on two primary text-driven human motion generation datasets: HumanML3D [2] and KIT Motion-Language (KIT-ML) [8]. We use the following five distinct metrics as evaluation metrics. 1) 'R-Precision' evaluates the accuracy of matching between text and motion; 2) 'Frechet Inception Distance (FID)' measures the similarity between the generated and ground

Generators	Evaluator	R-Precision (%) ↑		
		Top-1	Top-2	Top-3
Real motion	Default	51.1$^{\pm0.3}$	70.3$^{\pm0.3}$	79.7$^{\pm0.2}$
	CLaM	73.8$^{\pm0.2}$	87.0$^{\pm0.2}$	91.7$^{\pm0.1}$
MotionGPT [4]	Default	40.4$^{\pm0.2}$	56.7$^{\pm0.2}$	65.7$^{\pm0.1}$
	CLaM	47.8$^{\pm0.2}$	65.5$^{\pm0.2}$	75.2$^{\pm0.2}$
TM2T [3]	Default	42.4$^{\pm0.3}$	61.8$^{\pm0.3}$	72.9$^{\pm0.2}$
	CLaM	50.7$^{\pm0.2}$	67.9$^{\pm0.2}$	76.7$^{\pm0.2}$
T2M [2]	Default	45.5$^{\pm0.3}$	63.6$^{\pm0.3}$	73.6$^{\pm0.2}$
	CLaM	57.7$^{\pm0.3}$	73.0$^{\pm0.2}$	80.4$^{\pm0.2}$
MLD§ [1]	Default	48.1$^{\pm0.3}$	67.3$^{\pm0.3}$	77.2$^{\pm0.2}$
	CLaM	59.9$^{\pm0.3}$	76.0$^{\pm0.2}$	83.1$^{\pm0.2}$
MotionDiffuse§ [13]	Default	49.1$^{\pm0.1}$	68.1$^{\pm0.1}$	78.2$^{\pm0.1}$
	CLaM	64.5$^{\pm0.4}$	80.3$^{\pm0.3}$	86.8$^{\pm0.3}$
T2M-GPT [12]	Default	49.1$^{\pm0.3}$	68.0$^{\pm0.3}$	77.5$^{\pm0.2}$
	CLaM	67.6$^{\pm0.3}$	82.0$^{\pm0.4}$	87.8$^{\pm0.4}$

Table 2: Evaluation results on HumanML3D [2] test set with different evaluators. § denotes results using the ground-truth motion length as a precondition.

truth motion; 3) 'Diversity' measures the diversity of whole generated motion; 4) 'Multi Modality (MModality)' measures the diversity of generated motions within the same text description; 5) 'Multi-Modal Distance (MM-Dist)' measures the distance of motion feature and text feature. Due to space constraints, experiments on the KIT-ML are given in the open-source URL.

3.2 Evaluation with CLaM

Comparison on GT Motion Sequences. To demonstrate the validity of our CLaM evaluator, we compare the superior performance of our CLaM with the default evaluator [2] and retrievers [6, 7]. These retrievers treated the text-to-motion retrieval as the standalone task with the R-Precision metric. As shown in Table 1, we conduct a comparison of quantitative results based on ground-truth motion sequences. Our CLaM boasts the Top-1 R-Precision to 73.8% (+22.7%), a significant improvement over the original evaluator and retrievers.

Supported Text-to-Motion Algorithms. we evaluate the SoTA generators (e.g. TM2T [3], MLD [1]) using our CLaM evaluator and default evaluator [2] in Table 2. We notice that limited by the default evaluator, the performance of some methods is not accurately and distinctly measured, especially some of the SoTA methods actually have even better performance.

Ablation Studies. We analyze the influence of each part of our CLaM in Tab. 3. We observe a significant improvement in Top-1 R-Precision of 9.8% (64.0%⟶73.8%) when replacing the default evaluator with our CLaM model. The adaption of InfoNCE loss forces the optimization of latent spaces, with an improvement from 60.6% to 73.8%. Regardless of other conditions, the synonym provider plays an important role, reflecting the importance of the data scale.

3.3 HumanML3D-synthesis Dataset

As a by-product of our method, we combine the generated description sentences with the original description to create a new dataset called HumanML3D-synthesis. It includes 14,616 motions from the HumanML3D dataset and 547,102 descriptions made up of 11,506 different words. The stats for motion sequences are consistent with the HumanML3D dataset. The average and median lengths of the text descriptions are 12 and 11 words, respectively.

CLaM	SP	InfoNCE	R-Precision (%) ↑		
			Top-1	Top-2	Top-3
			51.1$^{\pm0.3}$	70.3$^{\pm0.3}$	79.7$^{\pm0.2}$
✓			53.1$^{\pm0.3}$	71.7$^{\pm0.2}$	81.1$^{\pm0.2}$
	✓		55.7$^{\pm0.2}$	74.0$^{\pm0.3}$	82.2$^{\pm0.2}$
✓	✓		60.6$^{\pm0.3}$	77.8$^{\pm0.3}$	84.9$^{\pm0.2}$
	✓	✓	64.0$^{\pm0.2}$	80.1$^{\pm0.2}$	87.2$^{\pm0.2}$
✓	✓	✓	**73.8**$^{\pm0.2}$	**87.0**$^{\pm0.2}$	**91.7**$^{\pm0.1}$

Table 3: Ablation studies of our CLaM model. 'SP' refers to the adoption of the synonym provider.

4 CONCLUSION

In this work, we introduce an open-source library featuring a robust Contrastive Language-and-Motion (CLaM) pre-training evaluator. This tool is designed to assess a range of text-driven human motion generation algorithms comprehensively. the effectiveness of our proposed evaluator is validated through extensive performance evaluations, utilizing metrics such as R-Precision. Furthermore, we launch the HumanML3D-synthesis dataset as a by-product, the largest of its kind, to further enrich the resources available in this field. These contributions pave the way for future development and application of mainstream text-driven human motion generation.

ACKNOWLEDGMENTS

This open-source work is sponsored by the Beijing Nova Program (NO. 20220484063).

REFERENCES

[1] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. 2023. Executing your Commands via Motion Diffusion in Latent Space. In *CVPR*. IEEE, 18000–18010.

[2] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. 2022. Generating Diverse and Natural 3D Human Motions from Text. In *CVPR*. IEEE, 5142–5151.

[3] Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. 2022. TM2T: Stochastic and Tokenized Modeling for the Reciprocal Generation of 3D Human Motions and Texts. In *ECCV*, Vol. 13695. Springer, 580–597.

[4] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. 2024. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems* 36 (2024).

[5] Mathis Petrovich, Michael J. Black, and Gül Varol. 2021. Action-Conditioned 3D Human Motion Synthesis with Transformer VAE. In *ICCV*. IEEE, 10965–10975.

[6] Mathis Petrovich, Michael J. Black, and Gül Varol. 2022. TEMOS: Generating Diverse Human Motions from Textual Descriptions. In *ECCV (22) (Lecture Notes in Computer Science, Vol. 13682)*. Springer, 480–497.

[7] Mathis Petrovich, Michael J Black, and Gül Varol. 2023. TMR: Text-to-motion retrieval using contrastive 3D human motion synthesis. In *ICCV*. 9488–9497.

[8] Matthias Plappert, Christian Mandery, and Tamim Asfour. 2016. The KIT Motion-Language Dataset. *Big Data* 4, 4 (dec 2016), 236–252.

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

[10] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *ACL (1)*. The Association for Computer Linguistics.

[11] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. 2023. Human Motion Diffusion Model. In *ICLR*.

[12] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. 2023. T2M-GPT: Generating Human Motion from Textual Descriptions with Discrete Representations. In *CVPR*.

[13] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. 2024. MotionDiffuse: Text-Driven Human Motion Generation with Diffusion Model. *TPAMI* (2024).