Motion Generation from Fine-grained Textual Descriptions With AI Feedback

1st Ruilin Wang
School of Computer Science and Engineering
Sun Yat-sen University
GuangZhou, China
wangrlin23@mail2.sysu.edu.cn

2nd Keze Wang*
School of Computer Science and Engineering
Sun Yat-sen University
GuangZhou, China
kezewang@gmail.com

***Abstract*—Text-to-motion generation aims to generate 3D human motion sequences that align with given textual descriptions. Most existing methods achieve remarkable performance under coarse-grained text conditions, but their applicability is limited in scenarios requiring fine-grained descriptions with rich temporal and spatial details. Prior studies on fine-grained text conditions leverage large language models(i.e., ChatGPT) to transform coarse-grained descriptions into fine-grained ones. But the contents usually include redundant, missing, or erroneous details due to the absence of visual input, reducing the text-motion alignment performance of the generative models. To address these challenges, we first investigate the performance of three mainstream generative models conditioned solely on the fine-grained descriptions. Secondly, to mitigate the above limitation of large language models, based on the group relative policy optimization(GRPO) algorithm, we propose a reinforcement learning framework to optimize the generated fine-grained descriptions with feedback from the specific evaluation models(AI Feedback). Specifically, we employ modality alignment scores as reward functions to steer the policy model toward generating fine-grained descriptions that enable the generative model to synthesize motions with higher fidelity and better text-motion alignment. Furthermore, we incorporate a contrastive mechanism: generated samples are compared against existing fine-grained annotations based on their rewards. Samples achieving rewards greater than or equal to the rewards of existing annotations are regarded as higher-advantage(positive reward) candidates within the group, whereas those receiving lower rewards are considered lower-advantage(negative reward) candidates. This mechanism explicitly guides the model's exploration toward more precise directions: surpassing existing fine-grained annotations, further enhancing the effectiveness of reinforcement learning. Extensive experiments on public datasets demonstrate the effectiveness of our approach. Notably, we observe that fine-grained descriptions generated by the large language model optimized in the proposed reinforcement learning framework provide consistent gains across the three generative models, highlighting the strong generalizability of our method.**

***Index Terms*—Text-to-motion, fine-grained descriptions, large language model, group relative policy optimization**

I. Introduction

The text-to-motion generation task involves generating 3D human motion sequences that align with given natural language descriptions. It has garnered significant attention due to its applications in animation production, virtual reality, and robotics. While recent cutting-edge studies have substantially advanced this field [1]–[9], most focus on generating sequences from coarse-grained text descriptions (containing only about 15 words). While such coarse-grained descriptions can represent the semantics of corresponding motion sequences, their lack of temporal and spatial information limits the capabilities of generative models when faced with complex descriptions. For instance, robotics-related operations involve textual descriptions rich in temporal (i.e., stepwise description) and spatial(i.e, joint transformation details) information.

FineHumanML3D [8] and FineMotion [9] leverage general large language models (e.g., ChatGPT [10] and Gemini [11]) to transform coarse-grained descriptions within existing motion datasets into fine-grained descriptions. However, these powerful large language models are inherently limited by the lack of visual input and strongly influenced by their linguistic priors. The generated fine-grained descriptions may fail to faithfully reflect the motion sequences, leading to two limitations: (i) missing or redundant details, and (ii) semantic misalignment with the actual motion, causing deviations from the original coarse-grained descriptions. These negatively impact the capabilities of models to learn text-motion alignment during training. Furthermore, although trained models are capable of learning mappings from fine-grained descriptions to motion, the richness of vocabulary and the diversity of syntactic structures in fine-grained descriptions can introduce substantial variability in the generated motions. During inference, descriptions that are semantically equivalent but differ in lexical choice or sentence structure may lead to noticeably difference in generated results. In addition, they explore the performance of text-to-motion models under fine-grained descriptions. However, both incorporate coarse-grained and fine-grained descriptions to provide sufficient semantic information, but obscure the impact of fine-grained descriptions on text-to-motion models, as numerous studies [1]–[7] have demonstrated that coarse-grained descriptions alone are enough for achieving high performance.

To address these limitations, we first investigate the training of text-to-motion models on motion data with only fine-grained annotation and systematically analyze the effectiveness of different model architectures under this setting. Subsequently, to alleviate the low text-motion alignment performance caused by imprecise or unfaithful descriptions, we

* Corresponding author

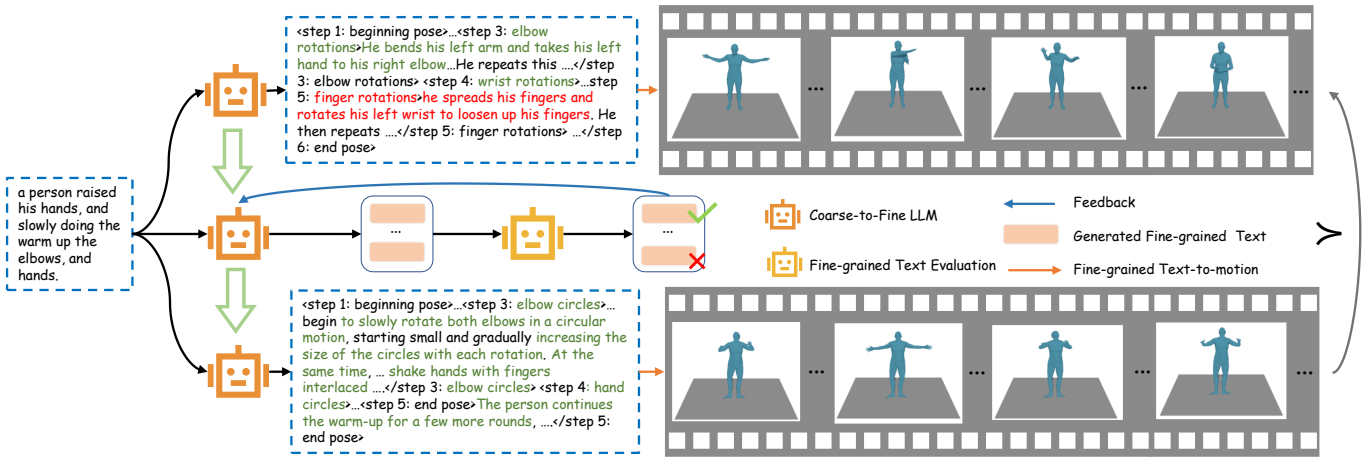

Fig. 1. Illustration of our approach's motivation. $\succ$ means strictly preferred over. Initially, the LLM generates a fine-grained description that uses imprecise terms (e.g., 'fingers') for hand actions, which can mislead the generation model and generate motions that deviate from the intended semantics. After reinforcement learning with negative-sample feedback, the model revises the description (e.g., 'hand'), leading to motions that more faithfully align with the original coarse-grained text.

optimize a large language model(LLM) to expand coarse-grained descriptions into fine-grained descriptions. Specifically, we train the LLM via supervised fine-tuning to learn the distribution of fine-grained descriptions used for training text-to-motion models. And apply the GRPO [12] algorithm to incorporate feedback from both the text-to-motion model and pre-trained text–motion alignment models. Through these two stages, we improve the faithfulness of the generated fine-grained descriptions, thereby enhancing the text-motion alignment performance of the generative models. Fig. 1 provides an illustrative overview of our motivation.

Unlike Motion-R1 [13] that directly computes advantages after obtaining continuous modality alignment scores, we incorporate a contrastive mechanism in which the reward scores of the annotated fine-grained descriptions in the original dataset serve as a baseline. The final reward is determined by comparison with this baseline: samples whose scores are greater than or equal to the baseline receive positive rewards, while those falling below it are assigned negative rewards. Through reinforcement learning, the model is encouraged to preferentially retain fine-grained descriptions whose performance is not inferior to the annotated fine-grained ones. Meanwhile, samples that suffer from issues including missing details, redundancy, or semantic mismatch are assigned negative rewards due to their inferior guidance performance. Thus, their output probabilities are suppressed. The detailed procedure of the proposed method is depicted in Fig. 2. Finally, we conducted extensive experiments on the FineHumanML3D [8] and HumanML3D [14] datasets to validate our method. Results demonstrate that, compared to fine-grained descriptions in the original dataset(annotated by ChatGPT [10]), descriptions generated by the LLM optimized by our proposed method can guide generative models to generate motions with higher text-motion alignment. Furthermore, we observe that even generative models that are not involved in reinforcement learning training can still benefit from the outputs of the optimized LLM, similarly enhancing their text-motion alignment performance, indicating the generalizability of our proposed method. Code is available at https://github.com/ruilin19/Fine-grained-Text-to-Motion-RL.

Our main contributions can be summarized as follows:

- We investigate the underexplored setting of text-to-motion generation conditioned solely on the fine-grained descriptions and analyze model performance in light of the characteristics of fine-grained descriptions.
- We propose to improve the performance of generation models by directly optimizing the fine-grained description conditions rather than the generation models themselves. Specifically, we introduce a reinforcement learning framework that optimizes the LLM to generate more faithful fine-grained descriptions from coarse-grained inputs. A contrastive mechanism is employed to explicitly encourage LLM to generate samples better than the original annotations during reinforcement learning.
- We conduct extensive experiments, and the results demonstrate that the descriptions generated by the optimized LLM lead to better text-motion alignment performance. Moreover, the descriptions also generalize well across different generation architectures, highlighting the effectiveness and generalizability of our approach.

II. Related Work

The text-to-motion (t2m) task aims to generate 3D human motion sequences aligned with given textual descriptions. Early works employ VAEs [15] to model motion representations with texts. With the advent of VQ-VAE [16], motion sequences can be quantized into discrete token sequences, enabling the application of language modeling techniques for this task. T2M-GPT [6] encodes textual descriptions using CLIP [17] and models the discrete motion tokens conditioned on text embedding in an autoregressive manner. MoMask [3] and BAMM [4] further demonstrate that masked modeling

and bidirectional modeling can effectively accomplish this task. In addition, MotionGPT [7], [18] leverages the text understanding capabilities of pretrained language models to perform conditional generation of discrete motion sequences. Diffusion models have also been shown to be effective for text-to-motion generation [5], [19]. However, due to limitations of current benchmarks [14], [20], most studies focus on generating motion sequences conditioned on coarse-grained descriptions, with limited exploration of scenarios requiring longer or fine-grained descriptions. FineMotion [9] and FineHumanML3D [8] investigate the T2M task using both coarse-grained and fine-grained descriptions, but do not analyze the impact of relying solely on the fine-grained descriptions.

Reinforcement learning has been shown to improve the performance of large language models by scaling sampled trajectories that satisfy reward signals [12], [21]. Several studies have explored its application to text-to-motion generation. [22] and [23] adopt DPO [21] to enhance generation quality while avoiding the high cost of manual annotation. Motion-R1 [13] demonstrates that incorporating modality-alignment rewards with the GRPO [12] can further improve model performance. These studies primarily optimize the generative model itself. PAE [24] and Dall·E 3 [25] suggest that directly optimizing the inputs to a generative model can achieve effects similar to optimizing the model parameters in enhancing higher-quality image generation. This strategy is particularly relevant when the condition is long and requires richer details, which is also applicable to text-to-motion generation under the fine-grained descriptions.

III. Methodology

See Fig. 2. Our method consists of three components: a motion tokenizer, a coarse-to-fine large language model(LLM), and a text-to-motion(t2m) model. The motion tokenizer maps continuous motion sequences into discrete motion token sequences, and conversely reconstructs continuous motion sequences from them. The coarse-to-fine LLM transforms coarse-grained descriptions into fine-grained ones enriched with temporal and spatial details. It also serves as the policy model during reinforcement learning. The t2m model generates discrete motion tokens or continuous motions conditioned on the fine-grained descriptions.

A. Motion Tokenizer

Similar to MoMask [3], we adopt Residual VQ-VAE [16] as the motion tokenizer. Two codebooks are employed to approximate the encoder's outputs, which effectively reduces quantization error and improves reconstruction quality. Specifically, the motion tokenizer consists of a motion encoder E, two codebooks $\mathcal{C}_1$ and $\mathcal{C}_2$, and a motion decoder D. The encoder E accepts motion sequences $M = [m_1, m_2, ..., m_L]$, $m_i \in \mathcal{R}^p$, p is the dimension of a single pose representation, M is then encoded into latent codes $\hat{Z} = E(M), \hat{Z} = [\hat{z}_1, \hat{z}_2, ..., \hat{z}_{L/r}]$, $\hat{z}_i \in \mathcal{R}^d$, d is the dimensionality of the encoder embedding space and r is the compression ratio of the encoder. The codebooks $\mathcal{C}_1, \mathcal{C}_2 \in \mathcal{R}^{512\times d}$ quantize the corresponding latent codes from encoders by searching the nearest embedding in them. Vector quantization operation is define as $\mathcal{Q}, \mathcal{S} = VQ(\hat{Z}, \mathcal{C}) = \{z_i^q, s_i \mid z_i^q, s_i = \arg\min_{z_k} \|z_k - \hat{z}_i\|_2, z_k \in \mathcal{C}, \hat{z}_i \in \hat{Z}, k \in [0, N), N \in \mathbb{Z}^+\}$,$s_i$ is the index of z_i^q in $\mathcal{C}$. Let $\mathcal{Q}_1$ and $\mathcal{S}_1$ be the quantized vector of the latent code $\hat{Z}$ and corresponding index sequences from $\mathcal{C}_1$. Let $\mathcal{Q}_2$ and $\mathcal{S}_2$ be the quantized vector of the residual latent code($\hat{Z} - Q_1$) and corresponding index sequences from $\mathcal{C}_2$, $\mathcal{Q}_1, \mathcal{S}_1 = VQ(\hat{Z}, \mathcal{C}_1)$,$\mathcal{Q}_2, \mathcal{S}_2 = VQ((\hat{Z} - Q_1), \mathcal{C}_2)$. The reconstructed motion sequences $\hat{M}$ is decoded by the decoder from the sum of $\mathcal{Q}_1$ and $\mathcal{Q}_2$, $\hat{M} = D(\mathcal{Q}_1 + \mathcal{Q}_2), \hat{M} = [\hat{m}_1, \hat{m}_2, ..., \hat{m}_L]$.

To optimize the Residual VQ-VAE, we introduce a reconstruction loss $\mathcal{L}_{rec}$ to minimize the reconstruction error between the ground truth motion M and the reconstructed one $\hat{M}$. In addition, we follow [6] to apply velocity regularization in the reconstruction loss:

$$\mathcal{L}_{rec} = \mathcal{L}_1^{smooth}(M, \hat{M}) + \alpha \mathcal{L}_1^{smooth}(V(M), V(\hat{M})), \quad (1)$$

$V(M)$ denotes extracting the velocity part from motion representations. α is used to control the degree of velocity regularization. For the optimization of codebooks, we follow [6] to apply exponential moving average (EMA) and codebook reset techniques [26] to update embeddings in the codebooks. Finally, we introduce a commit loss $\mathcal{L}_{commit}$ for the motion encoder to regularize the motion encoder, encouraging its outputs to remain close to the embeddings of corresponding codebooks.

$$\mathcal{L}_{commit} = \|sg(Q_1) - \hat{Z}\|_2 + \|sg(Q_2) - (\hat{Z} - sg(Q_1))\|_2, \quad (2)$$

where sg means stopping the gradient. In total, the training loss of motion tokenizer $\mathcal{L}_{vqvae}$ can be transformed to:

$$\mathcal{L}_{vqvae} = \mathcal{L}_{rec} + \gamma \mathcal{L}_{commit}, \quad (3)$$

where γ is a hyperparameter for commit loss.

B. Coarse-to-fine LLM

Inspired by Motion-R1 [13], we also leverage modality-aligned scores as rewards to provide feedback to the LLM, in order to mitigate redundancy or erroneous information in fine-grained descriptions generated by it. Since the original LLM is trained on general-domain training data, its output distributions exhibit high diversity, making it challenging to sample motion descriptions that are well-suited for the fine-grained text-to-motion task. This limitation can hinder efficient sampling by the policy model during rollouts, leading to inefficient and unstable exploration. To address this issue, we adopt a cold start training strategy. Specifically, we perform supervised fine-tuning on the LLM using training pairs composed of constructed prompts and coarse-to-fine description pairs from the original dataset. The data is formatted as follows:

Task Prompt:{*Coarse-grained Description*}
<*think*>{*Fine-grained Description*}</*think*>

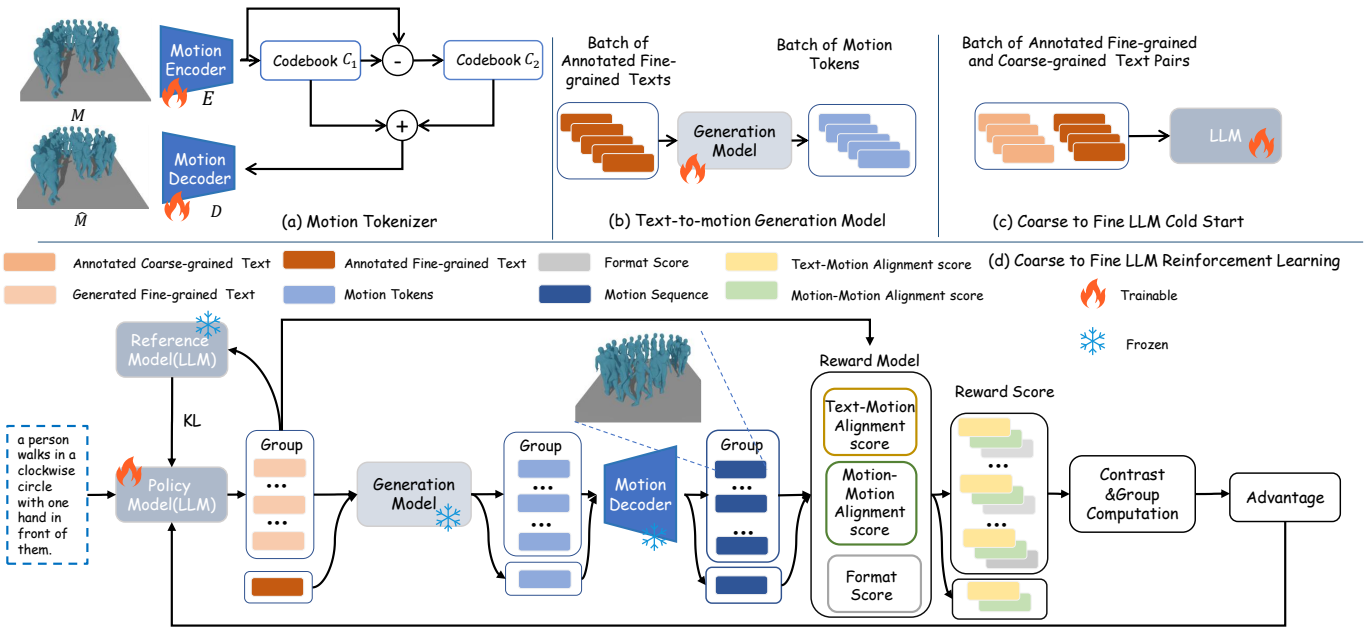

Fig. 2. Pipeline of our method. (a) We train a motion tokenizer to encode and reconstruct continuous motion sequences. (b) Using the discrete motion tokens and fine-grained descriptions from FineHumanML3D [8], we train a fine-grained text–to-motion generation model. (c) We train a large language model to transform coarse-grained descriptions from HumanML3D [14] to corresponding fine-grained descriptions in FineHumanML3D [8]. (d) This language model is then optimized via reinforcement learning to reduce redundancy and errors in the generated fine-grained descriptions.

Cold start training enables the LLM to learn the mapping from coarse-grained descriptions to fine-grained ones, aligning its output distribution closer to that of fine-grained ones. It provides a stable initial state for facilitating effective and consistent exploration in subsequent reinforcement learning.

Since supervised fine-tuning only approximates the distribution of ground-truth samples and lacks explicit supervision from negative samples, the model often produces outputs containing erroneous information during inference. To address this issue, we adopt the GRPO [12] algorithm, which improves the faithfulness of model outputs by providing both positive and negative feedback along generated trajectories. The GRPO [12] algorithm does not require additional supervised data. Instead, it updates model parameters by evaluating the policy model's sampled trajectories. Given a query q, the old policy $\pi_{\theta_{\text{old}}}$ samples a group of G candidate outputs, which are assigned rewards according to task-specific reward functions. The rewards within the group are then normalized to compute advantages, which are used to update the model parameters. The optimization objective is as follows:

$$\begin{aligned}&\mathcal{J}_{\text{GRPO}}(\theta)\\&= \mathbb{E}_{\{o_i\}\sim\pi_{\theta_{\text{old}}}}\Bigg[\frac{1}{G}\sum_{i=1}^{G}\min\Big(\rho_i A_i,\ \text{clip}(\rho_i, 1-\epsilon, 1+\epsilon)A_i\Big)\\&\quad-\beta\, D_{\text{KL}}(\pi_\theta\|\pi_{\text{ref}})\Bigg].\end{aligned} \tag{4}$$

$\rho_i = \frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}$ is the importance sampling ratio, ϵ is a hyper-parameter for controlling the clipping range, β is the weighting coefficient of the KL-divergence regularization term, π_θ is the current policy, π_{ref} is the reference policy, $A_i = \frac{r_i - mean(r)}{std(r)}$ denotes the group-wise advantage term, which is obtained by normalizing the aggregated rewards. It represents the advantage of each sample within the group. The advantages indicate whether a sampled trajectory is encouraged or suppressed, and thus the reward function directly influences the exploration direction of the policy model. Following Motion-R1 [13], we introduce a text–motion alignment reward and a motion–motion alignment reward to encourage the model to explore sampled trajectories that are semantically closer to the ground-truth texts and structurally closer to the ground-truth motion sequences. Additionally, to ensure that the policy model's sampled trajectories can be effectively utilized during exploration, we incorporate a format reward to constrain the output format. Detailed descriptions of these three reward functions are as follows:

Text–Motion Alignment Reward LLMs often generate fine-grained descriptions containing redundant, missing, or erroneous information, which can mislead the text-to-motion models into generating motions that are unfaithful to the original coarse-grained descriptions and motions. To mitigate this issue, we introduce the text–motion alignment score as the reward to encourage the LLM to generate more faithful descriptions. Specifically, the text–motion alignment score is defined as the cosine similarity in the embedding space between the generated motion sequence and the original coarse-grained descriptions. Both text and motion embeddings are obtained using pre-trained encoders from Guo [27].

Unlike question-answering tasks, where binary rewards can be directly assigned based on correctness. Cosine similarity

is a continuous value in the range of $[0, 1]$, so it is difficult to define a precise threshold to distinguish sample quality. To achieve a similar binary reward signal, we introduce a contrastive mechanism: the original fine-grained descriptions generated by the general large language model as baselines. By comparing the text–motion alignment scores of generated samples to those of the baselines, we can significantly differentiate the quality of samples within the group. Specifically, generated samples with scores lower than the baselines are more likely to contain erroneous information and are assigned negative rewards. Conversely, samples with scores exceeding the baseline are likely to be more faithful or have sentence structures and word choices similar to the training data distribution, thus they are assigned the positive rewards. This reward function is defined as follows:

$$R_{t2m} = \begin{cases} 1, & if\ sim(E_m(D(G(o))), E_t(T_{coarse})) \\ & \geq\ sim(E_m(D(G(T_{fine}))), E_t(T_{coarse})), \\ 0, & otherwise, \end{cases} \tag{5}$$

where sim is cosine similarity between embeddings, E_m and E_t denote the text and motion pre-trained encoders from Guo [27], T_{fine} and T_{coarse} denote the coarse-grained and fine-grained descriptions from the training dataset, o are samples from policy model, D and G denote the motion decoder from motion tokenizer and text-to-motion model.

Motion–Motion Alignment Reward We introduce a motion–motion alignment score as the reward to encourage the LLM to generate samples that guide the t2m model to generate motions closely aligned with the ground truth ones in both temporal and spatial characteristics. Specifically, the score is defined as the cosine similarity in the embedding space between the generated motions and the ground truth ones. Motion embeddings are also obtained using the pre-trained motion encoder from Guo [27]. Similar to the text–motion alignment score, we employ the contrastive mechanism to binarize the reward scores. The overall motion–motion alignment reward is defined as follows:

$$R_{m2m} = \begin{cases} 1, & if\ sim(E_m(D(G(o))), E_m(M)) \\ & \geq\ sim(E_m(D(G(T_{fine}))), E_m(m)), \\ 0, & otherwise, \end{cases} \tag{6}$$

where M is the ground truth motion from the training dataset.

Format Reward We introduce a format reward to ensure structured outputs from the policy model. Specifically, if the model's output successfully matches the target pattern string as follows, the reward is set to 1, otherwise, it is set to 0.

<think>{*Fine-grained Description*}*</think>*

These three rewards are aggregated to compute the advantages, guiding the policy model's exploration trajectories toward outputs that are both faithful and of high quality.

C. Text-to-Motion Generation Model

We reproduce three representative generative architectures for fine-grained text-to-motion task: the motion diffusion model(MDM) [5], T2M-GPT [6], and MotionGPT [18]. For MDM [5], we follow the default model configurations, using DistilBERT [28] as the text encoder, and the number of diffusion steps is set to 50. For T2M-GPT, due to the context window limitations of the CLIP [17], we replace it with T5-base [29]. For MotionGPT, we select T5-base as the backbone model and fine-grained text-to-motion as the only training task. The MotionGPT is renamed as FgT2M-T5.

TABLE I
TRAINING SETTING.

Hyper-parameter	Cold Start	Reinforcement Learning	FgT2M-T5
Epoch	40	1	300
Global Batch Size	128	8	32
Per-GPU Batch Size	4	2	32
Gradient Accumulation	8	1	1
Learning Rate	2e-5	1e-6	2e-4
Warm-up Ratio	0.02	0.02	0

Since the fine-grained descriptions contain significantly more tokens than coarse-grained ones, it requires the t2m model to extract motion-relevant information from a longer token sequence, thereby increasing the training difficulty. To address this challenge, we explore adopting block-wise local pooling over fine-grained text embeddings to reduce token-level complexity and investigate the performance of different t2m models under varying complexity of conditional inputs. We provide a detailed analysis in the experiment section.

IV. EXPERIMENT

A. Experiment Settings

Dataset We conduct experiments on HumanML3D [14] and FineHumanML3D [8]. HumanML3D [14] consists of 11,416 motions with manually annotated textual descriptions, ensuring high text–motion alignment quality. FineHumanML3D [8] augments HumanML3D by generating fine-grained descriptions from the original coarse-grained annotations using ChatGPT [10]. These annotations are noisier and may contain text–motion misalignment, redundancy, or missing information. We merge these two datasets and retain samples that include both coarse-grained and fine-grained descriptions for training and evaluation. The data split follows the standard protocol of HumanML3D.

Evalutaion Metric We adopts the *Frechet Inception Distance* (FID) [30], *Multi-modal Distance* (MM Dist), R-Precision Top3, and *Diversity* metric to assess the fidelity, diversity, and text-motion alignment of the generated motion. Each experiment is repeated 20 times, with reported measures representing mean values and 95% confidence intervals. Different from the evaluation protocol used in FineHumanML3D [8], during reinforcement learning and the evaluation of text–motion alignment metrics, we continue to use the original coarse-grained descriptions to compute text embeddings and text-motion similarity. Due to the redundancy or errors in the original fine-grained descriptions caused by the lack of visual input, some fine-grained annotations exhibit lower semantic alignment with the corresponding motion

TABLE II
COMPARISON OF FINE-GRAINED TEXT-TO-MOTION GENERATION ON HUMANML3D [14] TEST SET.

Method	FID↓	R-Precision Top3↑	MM Dist↓	Diversity→
Ground Truth	$0.002^{\pm.000}$	$0.797^{\pm.002}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$
FgT2M-T5 [18]	$0.178^{\pm.005}$	$0.766^{\pm.003}$	$3.147^{\pm.007}$	$9.830^{\pm.095}$
-Cold Start	$0.190^{\pm.006}$	$0.776^{\pm.003}$	$3.091^{\pm.011}$	$9.816^{\pm.072}$
-Ours	$\mathbf{0.176^{\pm.006}}$	$\mathbf{0.781^{\pm.002}}$	$\mathbf{3.068^{\pm.012}}$	$9.862^{\pm.086}$
-Ours w/o contrast	$0.177^{\pm.008}$	$0.779^{\pm.003}$	$3.075^{\pm.010}$	$\mathbf{9.776^{\pm.091}}$
MDM [5]	$\mathbf{0.515^{\pm.008}}$	$0.809^{\pm.002}$	$2.980^{\pm.008}$	$\mathbf{10.252^{\pm.099}}$
-Transfer*	$0.578^{\pm.009}$	$\mathbf{0.814^{\pm.002}}$	$\mathbf{2.944^{\pm.009}}$	$10.262^{\pm.087}$
T2M-GPT [6]	$0.183^{\pm.011}$	$0.721^{\pm.002}$	$3.465^{\pm.012}$	$\mathbf{9.607^{\pm.081}}$
-Transfer*	$\mathbf{0.175^{\pm.007}}$	$\mathbf{0.736^{\pm.002}}$	$\mathbf{3.362^{\pm.012}}$	$9.724^{\pm.086}$

* denotes that this model is not involved in the reinforcement learning.

compared to carefully annotated coarse-grained descriptions in HumanML3D. Applying such low-alignment texts to evaluate the text-motion alignment of the generated sample is biased.

Implementation Details We trained the Residual VQ-VAE following the same setting as T2M-GPT. For the t2m model, we use the default settings from MDM [5] and T2M-GPT [6]. For the coarse-to-fine LLM, we used Qwen-2.5 [31] 3B as the base LLM. Other training settings for coarse-to-fine LLM and FgT2M-T5 could be seen in Tab. I.

B. Main Results

As shown in Tab. II, fine-grained descriptions generated by the cold start model yield higher text-motion alignment than the original fine-grained ones(annotated by ChatGPT) when guiding the generation. Since the output distribution of the cold start model is better aligned with the training distribution of the t2m model, ChatGPT with a stronger expressive capacity may generate descriptions deviating from the training distribution, leading to degraded guidance performance. Applying our proposed reinforcement learning method improves both text-motion alignment and generation quality, demonstrating that modality alignment rewards effectively encourage the sampling of more faithful fine-grained descriptions. The observed performance drop after removing the contrastive mechanism highlights its role in providing more precise reward signals, thereby facilitating exploration toward a more optimal direction. Furthermore, to validate the generalization of the method, we adopt a transfer evaluation strategy which selecting the corresponding texts from the outputs of the LLM during the evaluation of FgT2M-T5 to replace those in the test set for testing. It further enhances text-motion alignment across MDM and T2M-GPT. Since they are not involved in reinforcement learning, results can be regarded as zero-shot evaluations, highlighting the generalization of the proposed method. Overall, all results indicate that generation performance can be improved by optimizing fine-grained descriptions alone.

C. Ablation Study

Pooling Size In this experiment, we explore block-wise local pooling to reduce the number of text tokens provided as conditional inputs. As shown in Tab. III, when applying FgT2M-T5, adopting the original length yields the best FID but the worst R-precision and MM Dist. It indicates that excessively long token sequences make it difficult for T5 to identify tokens representing key semantics, the decoder overfits the discrete motion tokens, resulting in high generation quality but low text-motion alignment. Then, the overfitting is alleviated by introducing block-wise local pooling, which concentrates local semantic information into fewer tokens, enabling the model to more effectively capture useful textual cues. However, a large local pooling size also hinders the learning of text–motion alignment. It is further evidenced by T2M-GPT, which pools the text embedding into a single one and exhibits degraded R-precision Top3 and MM Dist in Tab. II. In contrast, MDM achieves best performance when preserving the original length, likely due to its pose-level reconstruction objective: unlike discrete token prediction, reconstructing joint-level motion sequences is more challenging and benefits from retaining all token-level semantic information.

TABLE III
ABLATION STUDY OF TEXT EMBEDDING POOLING

Method	Pool Size	FID↓	R-Precision Top3↑	MM Dist↓	Diversity→
T5	1	$\mathbf{0.140^{\pm.007}}$	$0.731^{\pm.003}$	$3.403^{\pm.013}$	$9.678^{\pm.088}$
	4	$0.178^{\pm.005}$	$0.766^{\pm.002}$	$3.147^{\pm.007}$	$9.830^{\pm.095}$
	8	$0.189^{\pm.006}$	$\mathbf{0.771^{\pm.003}}$	$\mathbf{3.120^{\pm.011}}$	$9.790^{\pm.100}$
	16	$0.204^{\pm.006}$	$0.766^{\pm.003}$	$3.138^{\pm.010}$	$9.811^{\pm.083}$
MDM	1	$\mathbf{0.515^{\pm.008}}$	$\mathbf{0.809^{\pm.002}}$	$\mathbf{2.980^{\pm.008}}$	$10.252^{\pm.099}$
	8	$0.525^{\pm.009}$	$0.794^{\pm.002}$	$3.060^{\pm.010}$	$10.275^{\pm.096}$

Hyperparameters of GRPO We conduct ablation studies on the KL-divergence regularization weight and reward design under two configurations of the FgT2M-T5. As shown in Tab. IV, reducing the KL coefficient β from 0.01 to 0.001 improves both generation quality and text-motion alignment, indicating that a weaker constraint encourages more effective exploration. For reward design, using the text-motion alignment reward alone improves R-precision to a greater extent than FID. Conversely, the motion-motion alignment reward yields more substantial improvements in FID than in R-precision and MM Dist. These results align with the distinct optimization directions encouraged by these two rewards and suggest that they are complementary. When varying the generation model during reinforcement learning, the model with a local pooling size of 4 achieves the best overall performance across FID, R-precision, and MM Dist. We attribute this to smaller pooling blocks preserving more token embeddings, which increases the model's sensitivity to input conditions and provides the policy model with more diverse rewards during exploration, facilitating convergence toward more optimal trajectories.

D. Quality Results

We visualize motions generated under the guidance of the original fine-grained descriptions in FineHumanML3D [8] with those guided by fine-grained ones generated by our method. As depicted in Fig. 3. In the first example, both fine-grained descriptions contain key action terms (e.g., 'walk' and 'step on'). However, the original fine-grained description provides a brief description of the critical action of "stepping onto a big object", mentioning foot movements only three times.

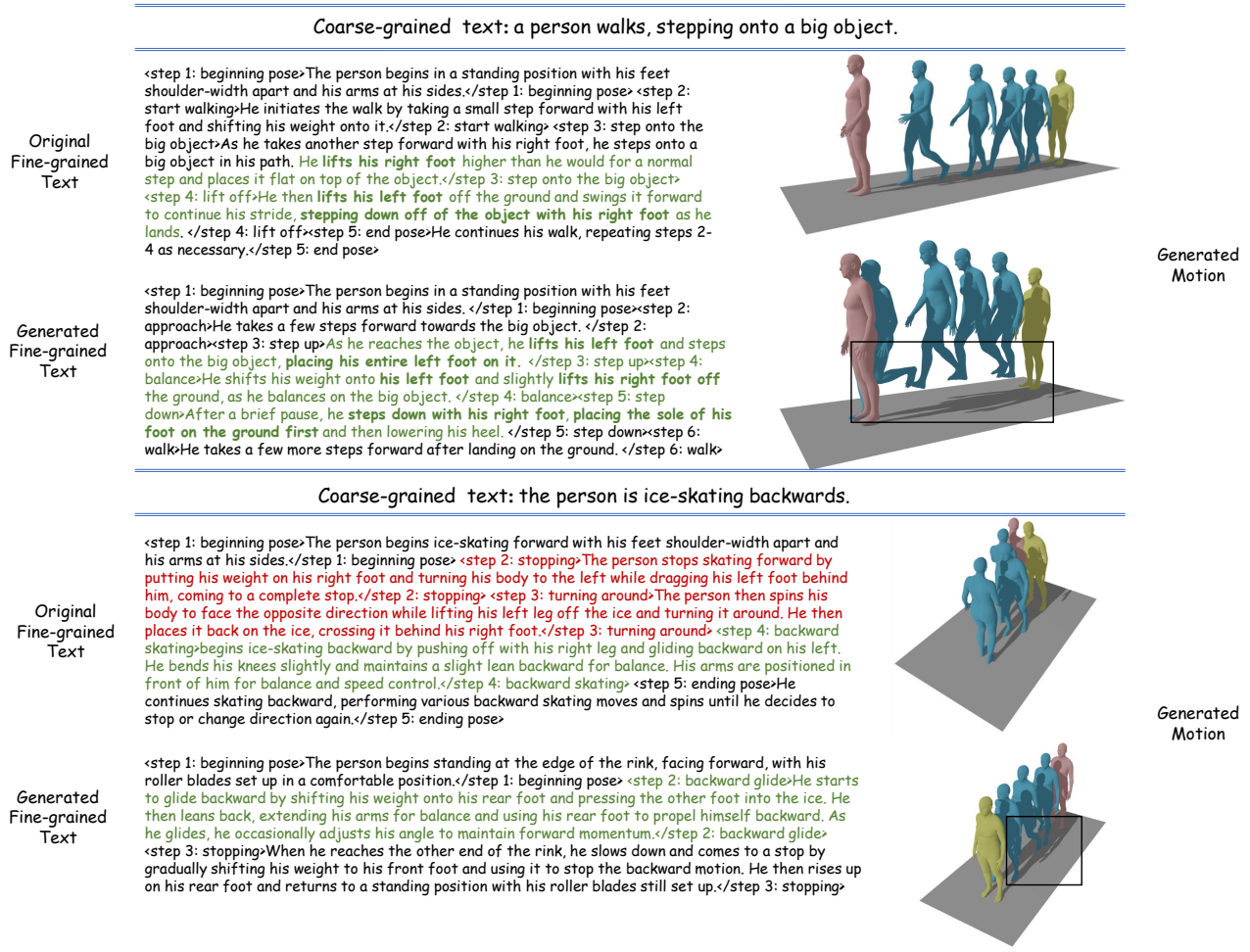

Fig. 3. Visualization. Yellow poses indicate the first frame of the sequence, and red poses indicate the last frame. Green text highlights key descriptions corresponding to the original coarse-grained text, while red text denotes erroneous or redundant descriptions.

TABLE IV
ABLATION STUDY OF GRPO HYPERPARAMETERS

SFT	beta	t2m	m2m	FID↓	R-Precision Top3↑	MM Dist↓	Diversity→
Pool Size = 4							
				$0.178^{\pm.005}$	$0.766^{\pm.002}$	$3.147^{\pm.007}$	$9.830^{\pm.095}$
✓				$0.190^{\pm.008}$	$0.776^{\pm.003}$	$3.091^{\pm.011}$	$9.816^{\pm.072}$
✓	0.01	✓	✓	$0.180^{\pm.008}$	$0.779^{\pm.003}$	$3.073^{\pm.009}$	$9.829^{\pm.068}$
✓	0.001	✓	✓	$\mathbf{0.176}^{\pm.006}$	$\mathbf{0.781}^{\pm.002}$	$\mathbf{3.068}^{\pm.012}$	$9.862^{\pm.086}$
✓	0.001	✓		$0.179^{\pm.007}$	$\mathbf{0.781}^{\pm.003}$	$3.070^{\pm.010}$	$9.863^{\pm.069}$
✓	0.001		✓	$\mathbf{0.176}^{\pm.006}$	$0.779^{\pm.002}$	$3.075^{\pm.010}$	$9.810^{\pm.098}$
Pool Size = 8							
				$\mathbf{0.189}^{\pm.006}$	$0.771^{\pm.003}$	$3.120^{\pm.011}$	$9.790^{\pm.100}$
✓				$0.219^{\pm.008}$	$0.775^{\pm.003}$	$3.099^{\pm.012}$	$9.776^{\pm.074}$
✓	0.001	✓	✓	$0.199^{\pm.009}$	$\mathbf{0.779}^{\pm.003}$	$\mathbf{3.077}^{\pm.008}$	$9.798^{\pm.095}$

This sparsity increases the difficulty for the t2m model to capture the corresponding semantic information within a long sequence, overlooking the action 'step on' during generation. In contrast, the generated fine-grained description includes six explicit references to foot movements, allocating more textual tokens to the 'step on' action. As a result, this semantic cue becomes more salient, enabling the t2m model to more reliably capture and reproduce the intended motion during generation.

In the second example, the original fine-grained description contains substantial redundant action cues, where the key semantic 'backwards' is underrepresented and further omitted by irrelevant actions such as 'turn around', leading the t2m model to neglect 'backward' movement in the generated motion. In contrast, the generated fine-grained one assigns a higher semantic weight to 'backwards', resulting in output that more faithfully aligns with the original coarse-grained. Overall, these observations indicate that our method systematically refines fine-grained descriptions by suppressing redundant or erroneous content, leading to more faithful conditioning signals and consistently improved generation performance.

V. CONCLUSION

In this paper, we first investigate the performance of three mainstream generative architectures when conditioned solely on the fine-grained descriptions. Specifically, we analyze the impact of the number of tokens in the text conditions on the performance of different generative models and subsequently make corresponding modifications to the model architecture.

Then, we propose a reinforcement learning framework to optimize a Large Language Model (LLM) tasked with transforming coarse-grained descriptions into fine-grained ones, aiming to mitigate redundancy and erroneous information in the output content. We adopt the modality alignment scores as the reward functions to evaluate different trajectories generated during the model's exploration. Furthermore, to provide more precise rewards and guide the model toward better exploration direction, we introduce a contrastive mechanism that utilizes the original ground-truth fine-grained descriptions as a baseline to determine the final rewards of different trajectories. Finally, we conduct extensive experiments to verify the effectiveness and generalizability of our proposed method.

VI. Acknowledgment

This work was supported in part by Fundamental Research Funds for the Central Universities, Sun Yat-sen University, under Grant 23hytd006 and 23hytd006-2. During the writing and revision of this paper, AI-assisted tools (ChatGPT and Qwen) were used to support linguistic polishing, grammatical proofreading, and the improvement of textual fluency and professionalism.

References

[1] Z. Su, M. Yin, F. Xie, P. Wu, B. Hua, and F. Zhan, "Ag-mask: Augmented 3d generative masked motion model for text-to-motion," in *2025 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2025, pp. 1–8.

[2] J. Fu, Y. Long, X. Wang, and J. Yin, "Llm-driven "coach-athlete" pretraining framework for complex text-to-motion generation," in *2024 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2024, pp. 1–7.

[3] C. Guo, Y. Mu, M. G. Javed, S. Wang, and L. Cheng, "Momask: Generative masked modeling of 3d human motions," *arXiv preprint arXiv:2312.00063*, 2023.

[4] E. Pinyoanuntapong, M. U. Saleem, P. Wang, M. Lee, S. Das, and C. Chen, "Bamm: bidirectional autoregressive motion model," in *European Conference on Computer Vision*. Springer, 2024, pp. 172–190.

[5] G. Tevet, S. Raab, B. Gordon, Y. Shafir, D. Cohen-or, and A. H. Bermano, "Human motion diffusion model," in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=SJ1kSyO2jwu

[6] J. Zhang, Y. Zhang, X. Cun, S. Huang, Y. Zhang, H. Zhao, H. Lu, and X. 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.

[7] Y. Zhang, D. Huang, B. Liu, S. Tang, Y. Lu, L. Chen, L. Bai, Q. Chu, N. Yu, and W. Ouyang, "Motiongpt: Finetuned llms are general-purpose motion generators," *arXiv preprint arXiv:2306.10900*, 2023.

[8] K. Li and Y. Feng, "Motion generation from fine-grained textual descriptions," in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024, pp. 11 625–11 641.

[9] B. Wu, J. Xie, M. Ding, Z. Kong, J. Ren, R. Bai, R. Qu, and L. Shen, "Finemotion: A dataset and benchmark with both spatial and temporal annotation for fine-grained motion generation and editing," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2025, pp. 13 837–13 846.

[10] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[11] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023.

[12] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li *et al.*, "Deepseekmath: Pushing the limits of mathematical reasoning in open language models," *arXiv preprint arXiv:2402.03300*, 2024.

[13] R. Ouyang, H. Li, Z. Zhang, X. Wang, Z. Zhu, G. Huang, and X. Wang, "Motion-r1: Chain-of-thought reasoning and reinforcement learning for human motion generation," *arXiv preprint arXiv:2506.10353*, 2025.

[14] C. Guo, S. Zou, X. Zuo, S. Wang, W. Ji, X. Li, and L. Cheng, "Generating diverse and natural 3d human motions from text," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 5152–5161.

[15] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[16] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[17] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," *arXiv preprint arXiv:2103.00020*, 2021.

[18] B. Jiang, X. Chen, W. Liu, J. Yu, G. Yu, and T. Chen, "Motiongpt: Human motion as a foreign language," *Advances in Neural Information Processing Systems*, vol. 36, pp. 20 067–20 079, 2023.

[19] X. Chen, B. Jiang, W. Liu, Z. Huang, B. Fu, T. Chen, and G. Yu, "Executing your commands via motion diffusion in latent space," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 18 000–18 010.

[20] M. Plappert, C. Mandery, and T. Asfour, "The KIT motion-language dataset," *Big Data*, vol. 4, no. 4, pp. 236–252, dec 2016. [Online]. Available: http://dx.doi.org/10.1089/big.2016.0028

[21] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn, "Direct preference optimization: Your language model is secretly a reward model," *Advances in neural information processing systems*, vol. 36, pp. 53 728–53 741, 2023.

[22] M. Pappa, L. Collorone, G. Ficarra, I. Spinelli, and F. Galasso, "Modipo: text-to-motion alignment via ai-feedback-driven direct preference optimization," *arXiv preprint arXiv:2405.03803*, 2024.

[23] X. Tan, H. Wang, X. Geng, and P. Zhou, "Sopo: Text-to-motion generation using semi-online preference optimization," *arXiv preprint arXiv:2412.05095*, 2024.

[24] W. Mo, T. Zhang, Y. Bai, B. Su, J.-R. Wen, and Q. Yang, "Dynamic prompt optimizing for text-to-image generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 26 627–26 636.

[25] J. Betker, G. Goh, L. Jing, T. Brooks, J. Wang, L. Li, L. Ouyang, J. Zhuang, J. Lee, Y. Guo *et al.*, "Improving image generation with better captions," *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, vol. 2, no. 3, p. 8, 2023.

[26] A. Razavi, A. Van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," *Advances in neural information processing systems*, vol. 32, 2019.

[27] C. Guo, S. Zou, X. Zuo, S. Wang, W. Ji, X. Li, and L. Cheng, "Generating diverse and natural 3d human motions from text," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5152–5161.

[28] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[29] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of machine learning research*, vol. 21, no. 140, pp. 1–67, 2020.

[30] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[31] QwenTeam, "Qwen2.5 technical report," 2025. [Online]. Available: https://arxiv.org/abs/2412.15115