GroundingGPT: Language Enhanced Multi-modal Grounding Model

Zhaowei Li[1,2], **Qi Xu**[1], **Dong Zhang**[2], **Hang Song**[1], **Yiqing Cai**[1], **Qi Qi**[1], **Ran Zhou**[1], **Junting Pan**[1], **Zefeng Li**[1], **Van Tu Vu**[1], **Zhida Huang**[1], **Tao Wang**[1]
[1]ByteDance Inc, [2]Fudan University
lizhaowei126@gmail.com
https://lzw-lzw.github.io/GroundingGPT.github.io/

Abstract

Multi-modal large language models (MLLMs) have demonstrated remarkable performance across various tasks. However, these models often prioritize capturing global information and overlook the importance of perceiving local information. This limitation hinders their ability to effectively understand fine-grained details and handle grounding tasks that necessitate nuanced comprehension. Although some recent works have made strides in this, they have primarily focused on single-modality inputs. Therefore, we propose **GroundingGPT**, an end-to-end language enhanced multi-modal grounding model. It is designed to perform fine-grained grounding tasks for three modalities: image, video and audio. To enhance the model's performance, we adopt a coarse-to-fine training strategy, utilizing a three-stage training approach to progressively enhance the model's semantic awareness and fine-grained understanding capabilities. Additionally, we employ a diversified stage-specific dataset construction pipeline, developing a multi-modal, multi-granularity dataset tailored for training the model in different stages. Extensive experiments conducted on multiple multi-modal benchmarks demonstrate that our model achieves impressive fine-grained understanding of multi-modal inputs on grounding tasks while maintaining or improving its global comprehension capabilities. Our code, model, and dataset are available at https://github.com/lzw-lzw/GroundingGPT.

1 Introduction

Building upon the capabilities of large language models (LLMs), research on multi-modal large language models (MLLMs) has also advanced, enabling understanding across a broader range of modalities. Representative models such as LLaVA (Liu et al., 2023a) and MiniGPT-4 (Zhu et al., 2023) align visual features obtained from image encoders with LLM embedding space through visual instruction tuning, facilitating tasks such as image captioning and visual question answering.

However, existing MLLMs primarily focus on capturing global information while neglecting the fine-grained local information in multi-modal inputs. This limitation restricts their applicability in grounding tasks requiring a more detailed understanding. Shikra (Chen et al., 2023b), BuboGPT (Zhao et al., 2023) and Ferret (You et al., 2023) have explored techniques that enable finer alignment and understanding of inputs. By considering local-level information, these models exhibit enhanced performance in grounding or referring tasks. These methods provide insights into fine-grained understanding, but they are primarily limited to a single modality. There is still significant potential for exploring fine-grained understanding across other modalities.

To address the aforementioned issue, this paper proposes **GroundingGPT**, a language enhanced multi-modal grounding model, which is an end-to-end unified large language model designed to perform multi-modal grounding and understanding tasks across various modalities, including image, video, and audio. The comparison between our model and other models can be found in Table 1. Specifically, our model employs modality-specific adapters to map feature representations from individual encoders to the embedding space of LLMs. To incorporate spatial and temporal information, we directly represent coordinates and timestamps as textual numbers, eliminating the need for vocabulary expansion. For training GroundingGPT, we design a three-stage coarse-to-fine training strategy. In the first stage, we align each pre-trained multi-modal encoder with the LLM embedding space using modality-specific adapters. In the second stage, we aim to enable the model to capture fine-grained information, including coordinates and timestamps. In the third stage, we perform multi-granularity instruction tuning to refine the model's

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6657–6678

Models	Image Grounding	Video Grounding	Audio Grounding	Multi-turn Dialog	E2E
LLaVA	✗	✗	✗	✓	✓
Video-LLaMA	✗	✗	✗	✓	✓
Shikra	✓	✗	✗	✗	✓
Ferret	✓	✗	✗	✓	✗
BuboGPT	✓	✗	✓	✓	✗
LLaVA-Grounding	✓	✗	✗	✓	✗
GroundingGPT	✓	✓	✓	✓	✓

Table 1: Comparison of multi-modal large language models. "Multi-turn Dialog" refers to the model's ability to engage in multi-turn conversations with users. "E2E" refers to the models that are designed to be end-to-end architecture without the need for external modules.

responses. For each stage, we employed a stage-specific dataset construction pipeline to generate a diverse, multi-modal, and multi-granularity training dataset.

To summarize, our contributions are as follows:

- We propose GroundingGPT, an end-to-end multi-modal grounding model that accurately comprehends inputs and possesses robust grounding capabilities across multi modalities, including image, video and audio. To the best of our knowledge, GroundingGPT is the first model to achieve multi-modal fine-grained understanding and grounding.
- For training GroundingGPT, we employ a three-stage coarse-to-fine training process that enables the model to capture high-level semantic information and low-level fine-grained details simultaneously. To address the issue of limited data, we construct a diverse and high-quality multi-modal training dataset, which comprises a rich collection of multi-modal data enriched with fine-grained information.
- Extensive experiments conducted on a wide range of MLLM benchmarks demonstrate the generality and effectiveness of GroundingGPT in multi-modal grounding and understanding tasks across various modalities.

2 Related Work

Multi-modal Large Language Models (MLLMs) Recently, large language models (LLMs) represented by GPTs (Brown et al., 2020; OpenAI, 2023) and LLaMA (Touvron et al., 2023) have received extensive attention from researchers for their remarkable performance in various natural language processing tasks. Substantial progress has been made in the field of MLLMs, which extend the support for multi-modal input and output beyond language. These MLLMs typically fine-tune pre-trained LLMs with multi-modal instructions, to enable understanding across multiple modalities. Models such as LLaVA, MiniGPT-4, and mPLUG-Owl (Ye et al., 2023) map image embeddings obtained from image encoders into the LLM space. Similarly, video MLLMs like Video-Chat (Li et al., 2023b), Video-LLaMA (Zhang et al., 2023c), Video-Chatgpt (Maaz et al., 2023) and Valley (Luo et al., 2023), as well as speech MLLMs like SpeechGPT(Zhang et al., 2023b) and LLaSM (Shu et al., 2023), acquire multi-modal understanding capabilities through similar approaches. In X-LLM (Chen et al., 2023a), each modality is processed independently through dedicated branches for multi-modal input processing. Pandagpt (Su et al., 2023) employs a unified embedding space trained by ImageBind (Girdhar et al., 2023) to facilitate joint understanding of various modal inputs. However, these models often fail to adequately capture details within inputs.

MLLMs For Grounding Task Recently, there has been a focus on training visual MLLMs to achieve fine-grained image understanding and visual grounding. Approaches such as KOSMOS-2 (Peng et al., 2023) and Shikra achieve this by incorporating coordinates into the training data, enabling MLLMs to understand the location within images. On the other hand, approaches like NExT-Chat (Zhang et al., 2023a), LLaVA-grounding (Zhang et al., 2023d),GlaMM (Rasheed et al., 2023) and Ferret enhance perception of fine-

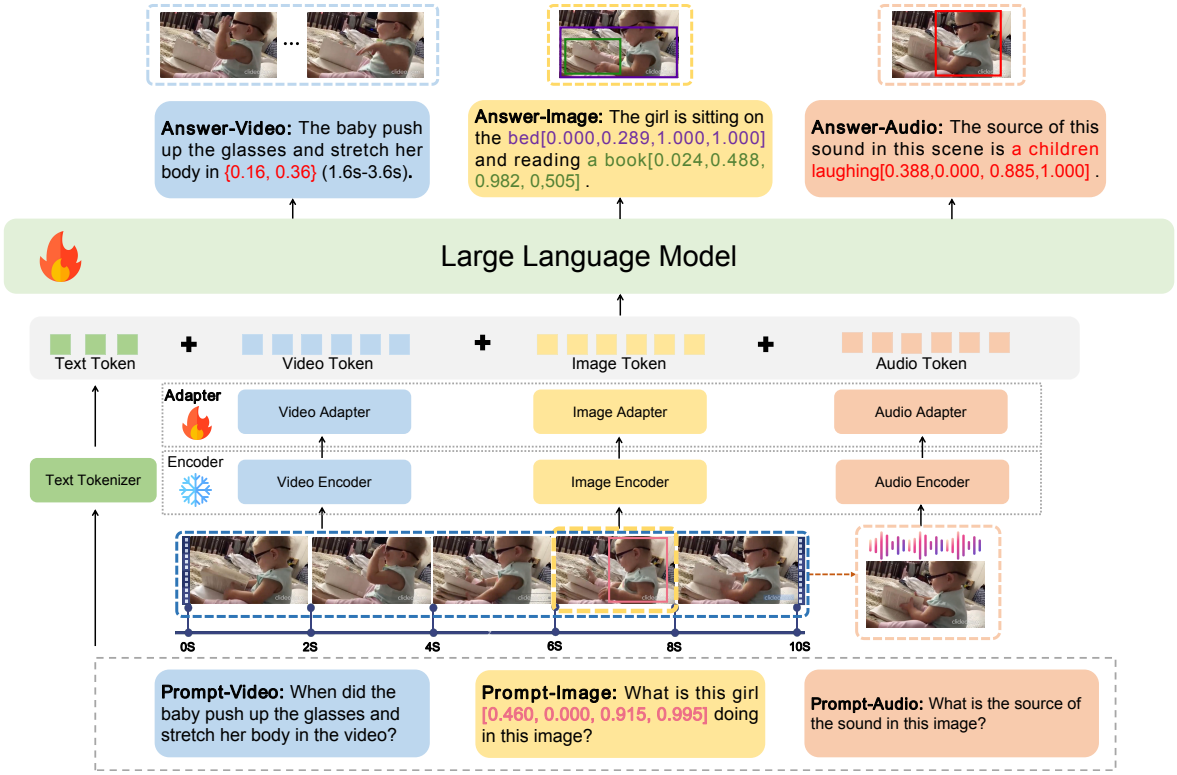


Figure 1: The overall structure of GroundingGPT involves separate encoders and adapters for each modality. Blue boxes represent video inputs, yellow boxes represent image inputs, and pink boxes represent audio inputs.

grained information by introducing additional region encoder modules. VTimeLLM (Huang et al., 2023) demonstrates the capability to understand fine-grained video moment and reason with respect to time boundary. BuboGPT (Zhao et al., 2023) enables cross-modal interaction between image, audio, and language, facilitating fine-grained understanding of different modalities.

3 Methods

We introduce the overall architecture of the GroundingGPT model in this section. Additionally, we will present our three-stage coarse-to-fine training strategy and data construction pipeline.

3.1 Model Architecture

Figure 1 illustrates the overall architecture of the GroundingGPT model. Multi-modal inputs are processed through modality-specific encoders to extract features. These features are then mapped to the LLM embedding space using corresponding adapters. We will also introduce the representation of coordinates and timestamps.

3.1.1 Image Branch

We employ the pre-trained CLIP visual encoder ViT-L/14 (Radford et al., 2021) to extract image features. The encoded image is represented as a fixed-length embedding vector $I \in R^{K_I \times d_I}$. To align the image representation with the LLM embedding space, we use an MLP to map the obtained features to the dimensions of LLMs. The mapped embeddings are then concatenated with text embeddings and used as input to LLMs, similar mapping methods are adopted for other modalities.

3.1.2 Video Branch

Considering the inherent information redundancy in videos and memory limitations, we uniformly sample M frames form the video. Each frame is processed by the image encoder, resulting in $V_f = [v_1, v_2, \ldots, v_M]$ where $v_i \in R^{K_f \times d_f}$ represents the embedding of the i-th frame. To preserve temporal information, we introduce temporal position encoding to the representation. The enhanced representation is then fed into the Video Q-former with the same structure as the Q-Former in BLIP-2 (Li et al., 2023a) to aggregate video information, which generates k_V video embedding vectors of dimensions d_V. These vectors form the representa-

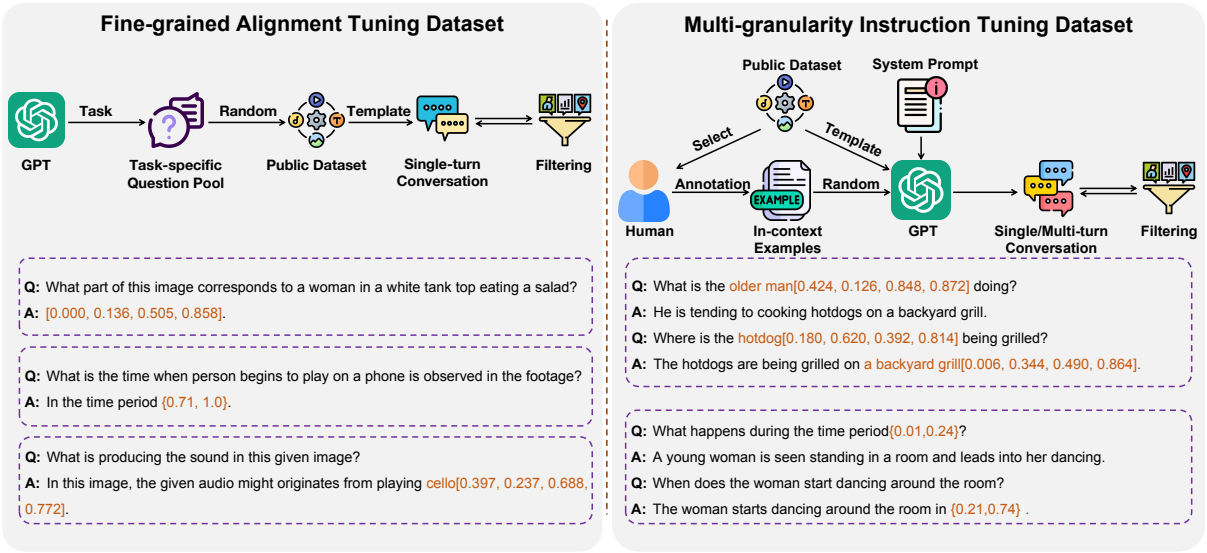


Figure 2: The data construction pipeline and examples for the last two training stages. To simplify, the multi-turn conversation examples only showcase two rounds of question-answer interactions.

tion $V \in R^{k_V \times d_V}$ for the entire video.

3.1.3 Audio Branch

The audio branch follows a structure similar to the video branch. We employ the ImageBind audio encoder, which processes 2-second audio clips with a 16kHz sampling rate and converts them into spectrograms using 128 mel-spectrogram bins. We sample N 2-second segments from the original audio and transform each segment into a vector, resulting in $A_s = [a_1, a_2, \ldots a_N]$, where $a_i \in R^{K_s \times d_s}$ represents the embedding of the i-th aduio segment. We incorporate temporal position encoding into A_s. Finally, we obtain a fixed-length audio representation sequence denoted as $A \in R^{k_A \times d_A}$ using the audio Q-former like video branch.

3.1.4 Spatial-temporal Representation

We represent the bounding box in an image using four relative coordinate values: $[x_1, y_1, x_2, y_2]$. These values correspond to the upper left corner point and the lower right corner point of the bounding box. Each value is rounded to three decimal places. We concatenate this textual representation after the description related to the bounding box. Similarly, for representing timestamps, we use two two-digit decimals $\{t_1, t_2\}$ to indicate the relative values of the start and end times of a time segment with respect to the total duration. This representation allows us to train the model without requiring additional vocabulary expansion or training. Examples of the training dataset are shown in Figure 2.

3.2 Coarse-to-Fine Training and Dataset

We employ a three-stage coarse-to-fine training strategy to train the model, while constructing specific datasets for each stage.

3.2.1 Multi-modal Pre-training

This stage focus on enabling the model to comprehend multi-modal inputs and develop a high-level semantic perception of the input. During the training process, the LLM and the encoders for each modality remain frozen, while only the adapters for each modality are trained.

Training Dataset We utilize public pretraining datasets as the primary source of our data. The training data for the image and video modalities is LLaVA-Pretrain-595k and Valley-Pretrain-703k, respectively. To construct the audio data, we adopt a similar approach as in LLaVA, leveraging the Wavcaps (Mei et al., 2023) dataset. Each sample is accompanied by a sampled instruction that requires the model to provide a concise description of the audio to construct a single-turn conversation.

3.2.2 Fine-grained Alignment Tuning

The second stage aims to enable the model to comprehend more detailed information, including coordinates and timestamps. Through training in this stage, the model achieves impressive results in various grounding tasks, establishing a more comprehensive and refined understanding ability. During

the training process, the encoders for each modality are frozen, while the LLM and adapters are trained.

Training Dataset The training data used in this stage includes the spatial-temporal representation mentioned in Section 3.1.4. To address the scarcity of fine-grained multi-modal data, we construct a multi-modal dataset specifically designed for this stage. The dataset is primarily obtained by converting publicly available datasets. As depicted in the left part of Figure 2, task descriptions are provided to GPT-3.5 to generate a task-specific question pool. For each data sample, a question is randomly selected from the pool, and templates are used to convert the sample's format, resulting in a single-turn conversation. For the image modality, we utilize visual grounding datasets such as RefCOCO (Kazemzadeh et al., 2014), RefCOCO+ (Kazemzadeh et al., 2014), RefCOCOg (Mao et al., 2016) and Visual Genome (Krishna et al., 2017b) to construct the datasets. For the video modality, video temporal grounding datasets such as DiDeMo (Anne Hendricks et al., 2017), HiREST (Zala et al., 2023) are utilized for fine-grained alignment. Regarding the sound localization task, we employ the VGGSS (Chen et al., 2021) dataset for training. All these datasets are transformed into single-turn conversation format following the aforementioned pipeline for training.

3.2.3 Multi-granularity Instruction Tuning

After the training in the first two stages, the model has acquired a strong understanding and grounding capability. This stage aims to enable the model to generate responses that better align with human preferences and improve multi-modal interactions. We train the model using instruction-tuning datasets at different granularities. Similar to the second stage, the encoders for each modality are frozen, while the LLM and adapters are trained.

Training Dataset The data utilized in this stage consists of high-quality fine-grained instruction-tuning dataset we construct and public instruction-tuning dataset. As illustrated in the right part of Figure 2, we select a subset of public datasets for human annotation to create in-context examples. It assists in guiding GPT-3.5 to follow similar patterns when generating instruction-tuning dataset. Subsequently, task-specific system prompts and randomly selected examples are input to GPT-3.5 to generate single/multi-turn conversations. For the image modality, we construct fine-grained datasets using the Flickr30K Entities (Plummer et al., 2015) dataset, including detailed descriptions and conversations. To enhance the model's fine-grained reasoning capability, we utilize the VCR (Zellers et al., 2019) dataset to construct a reasoning dataset with coordinates. For the video modality, we constructed datasets with temporal information by incorporating datasets from various video tasks such as DiDeMo (Anne Hendricks et al., 2017) and Activitynet Captions (Krishna et al., 2017a), along with other relevant sources. The public instruction-tuning datasets we use include LLaVA-v1.5-mix665k, Valley-Instruct-73k, Videochat-Instruct-11k, and an audio instruction-tuning dataset constructed using Clotho (Drossos et al., 2020) dataset. For more details about the datasets, please refer to appendix B.

To ensure the quality of the dataset, we carefully filter the data by eliminating samples that do not conform to the desired format or criteria. Specifically, we performed data cleaning on the raw data, which entailed filtering the downloaded data and removing any damaged instances, particularly videos with corrupted content. By ensuring data integrity, we preserved the quality of the dataset. Furthermore, during the data generation process, although we provided contextual examples, there were instances where the generated samples deviated from the desired format outlined in Section 3.1.4. For example, there were cases where the parentheses in coordinate representations did not match. To address this issue, we employed a set of predefined regular expression patterns to filter out samples that did not conform to the specified format.

During training, in order to prevent catastrophic forgetting in subsequent training stages, we adopt a sampling strategy that incorporates training data from previous stages. The training process employs a consistent training objective as follows:

$$L(\theta) = -\mathbb{E}_{(x,y)\sim D_{\text{current}}}[\log p(y|x)] - \alpha \cdot \mathbb{E}_{(x,y)\sim D_{\text{previous}}}[\log p(y|x)],$$

where $D_{current}$ denotes the dataset in current training stage, $D_{previous}$ denotes the dataset in previous training stage and α denotes the sampling rate. In the first training stage, α is set to 0.

4 Experiments

4.1 Experimental Setup

We employ Vicuna-v1.5 (Chiang et al., 2023) as the language model. Each training stage lasts for

Models	LLM Size	RefCOCO			RefCOCO+			RefCOCOg	
		val	testA	testB	val	testA	testB	val	test
UNITER	-	81.41	87.04	74.17	75.90	81.45	66.70	74.02	68.67
MDETR	-	86.75	89.58	81.41	79.52	84.09	70.62	81.64	80.89
UniTAB	-	86.32	88.84	80.61	78.70	83.22	69.48	79.96	79.97
KOSMOS-2	1.6B	52.32	57.42	47.26	45.48	50.73	42.24	60.57	61.65
Shikra	7B	87.01	90.61	80.24	81.60	87.36	72.12	82.27	82.19
NExT-Chat*	7B	85.50	90.00	77.90	77.20	84.50	68.00	80.10	79.80
Ferret*	7B	87.49	91.35	82.45	80.78	**87.38**	73.14	**83.93**	**84.76**
GroundingGPT	7B	**88.02**	**91.55**	**82.47**	**81.61**	87.18	**73.18**	81.67	81.99

Table 2: Performance comparison on the referring expression comprehension(REC) task. "*" indicates that the model employs additional image region perception modules.

Models	Charades-STA	
	R@1(IoU=0.5)	R@1(IoU=0.7)
Video-LLaMA	3.8	0.9
VideoChat	3.3	1.3
VideoChatGPT	7.7	1.7
GroundingGPT	**29.6**	**11.9**

Table 3: Performance comparison on the temporal grounding task. All the models have the same LLM size of 7B.

one epoch. During the training process, all images were padded to a square shape and resized to a resolution of 336×336. For each video, 64 frames were sampled, and for each audio, three 2-second segments were sampled and processed. For more details on the hyper-parameter settings, please refer to the appendix A.

4.2 Quantitative Evaluation

We conducted extensive experiments for the effectiveness of GroudingGPT in multi-modal grounding and understanding tasks.

4.2.1 Multi-modal Grounding

In this section, we demonstrate that our model achieves impressive fine-grained understanding of multi-modal inputs on grounding tasks.

Image Grounding In the image grounding task, the model takes an image and textual description of a region within the image as input, and outputs the text along with explicit position information, such as bounding boxes or masks. To assess the image grounding capability of the GroundingGPT model, we conduct experiments on the widely used Reference Expression Understanding (REC) task. The REC task requires the model to locate the bounding box corresponding to a given text reference expression. Our experiments involve three datasets: RefCOCO, RefCOCO+ and RefCOCOg. The baselines used for comparing include previous end-to-end multi-modal models UNITER (Chen et al., 2020), MDETR (Kamath et al., 2021), UniTAB (Yang et al., 2022), and the LLM-based multi-modal grounding models KOSMOS-2, Shikra, NExT-Chat and Ferret. For GroundingGPT model, we use a unified prompt like "Output the coordinate of <exp>", where "<exp>" represents the reference expression. The results on the REC task is presented in Table 2. GroundingGPT demonstrates remarkable performance across multiple datasets and performs comparably to specialized fine-tuned models or MLLMs that incorporate additional image region perception modules.

Video Grounding Video grounding focuses on identifying and localizing specific moments or events in the video based on given descriptions. To evaluate the video grounding capability of GroundingGPT, we conduct experiments on the temporal video grounding task. The temporal video grounding task requires models to pinpoint and highlight temporal boundaries within videos, that corresponds accurately to a specified text query.For the task, we employed datasets from Charades-STA (Gao et al., 2017). The predicted time segments are compared with the corresponding ground truth time segments to calculate the IoU. The evaluation metric used is "R@1, IoU = m", which mea-

Models	LLM Size	VQA^{v2}	GQA	VisWiz	SQA^{I}	VQA^{T}	POPE	MME	MMB	LLaVA^{W}
BLIP-2	13B	41.0	41	19.6	61	42.5	85.3	1293.8	-	38.1
InstructBLIP	7B	-	49.2	34.5	60.5	50.1	-	-	36	60.9
InstructBLIP	13B	-	49.5	33.4	63.1	50.7	78.9	1212.8	-	58.2
Shikra	13B	77.4	-	-	-	-	-	-	58.8	-
LLaVA-1.5	7B	78.5	62.0	50.0	66.8	**58.2**	85.9	**1510.7**	**64.3**	63.4
GroundingGPT	7B	**78.7**	**62.1**	**55.1**	**78.7**	55.2	**87.4**	1454.2	63.8	**70.9**

Table 4: Comparison of MLLMs on image understanding benchmarks. Benchmark names are abbreviated due to space limits. VQA-v2 (Goyal et al., 2017); GQA (Hudson and Manning, 2019); VisWiz (Gurari et al., 2018); SQA^{I}:ScienceQA-IMG (Lu et al., 2022); VQA^{T}: TextVQA (Singh et al., 2019); POPE (Li et al., 2023c); MME (Fu et al., 2023); MMB:MMBench (Liu et al., 2023b); LLaVA^{W}: LLaVA-Bench (In-the-Wild) (Liu et al., 2023a).

Models	LLM Size	MSVD-QA		MSRVTT-QA		ActivityNet-QA	
		Accuracy	Score	Accuracy	Score	Accuracy	Score
VideoChat	7B	56.3	2.8	45.0	2.5	26.5	2.2
Video-LLaMA	7B	51.6	2.5	29.6	1.8	12.4	1.1
Video-ChatGPT	7B	64.9	3.3	49.3	2.8	35.2	2.7
Valley	7B	65.4	3.4	45.7	2.5	42.9	3.0
GroundingGPT	7B	**67.8**	**3.7**	**51.6**	**3.1**	**44.7**	**3.2**

Table 5: Comparison of MLLMs on video understanding benchmarks. We adopt the evaluation methodology in Video-ChatGPT (Maaz et al., 2023) for evaluation.

Models	LLM Size	Random			Popular			Adversarial		
		Accuracy	F1-Score	Yes	Accuracy	F1-Score	Yes	Accuracy	F1-Score	Yes
LLaVA	7B	72.16	78.22	76.29	61.37	71.52	85.63	58.67	70.12	88.33
mPLUG-Owl	7B	53.97	68.39	95.63	50.90	66.94	98.57	50.67	66.82	98.67
MiniGPT-4	13B	79.67	80.17	52.53	69.73	73.02	62.20	65.17	70.42	67.77
InstructBLIP	13B	88.57	**89.27**	56.57	82.77	84.66	62.37	72.10	77.32	73.03
Shikra	7B	86.90	86.19	43.26	83.97	83.16	45.23	83.10	82.49	46.50
GroundingGPT	7B	**89.79**	89.22	**43.13**	**88.23**	**87.38**	**43.23**	**86.17**	**85.50**	**45.43**

Table 6: Results on the POPE benchmark for object hallucination evalaution. "Yes" represents the probability of positive answers to the given question.

sures the percentage of correctly retrieved moments with an IoU greater than m. We set the values of m as 0.5, 0.7 to assess different levels of accuracy. As shown in Table 3, GroundingGPT exhibits excellent performance in temporal video grounding task compared to previous video MLLMs, which primarily focuses on entire video understanding.

4.2.2 Multi-modal Understanding

We validate that GroundingGPT can maintain or improve the multi-modal understanding ability by introducing grounding tasks. Especially, it can effectively suppress object hallucination.

Image Understanding We evaluate the image understanding capability of GroundingGPT on five question-answering benchmarks and four recent proposed benchmarks specifically designed for vision instruction tuning. These benchmarks provide a comprehensive assessment of the model's capabilities using diverse evaluation metrics. The experimental results presented in Table 4 demonstrate that GroundingGPT achieves state-of-the-art performance on six benchmarks and remains highly competitive on other three benchmarks. Additionally, GroundingGPT exhibits advanced capabilities compared to larger-scale grounding MLLMs, such

Figure 3: Qualitative results of GroundingGPT on multi-modal grounding tasks.

as Shikra-13B.

Video Understanding In Table 5, we provide a quantitative assessment of the video question answering capabilities of MLLMs on three datasets: MSVD-QA (Chen and Dolan, 2011), MSRVTT-QA (Xu et al., 2016) and ActivityNet-QA (Yu et al., 2019). GroundingGPT consistently outperforms other baselines, demonstrating its advanced video understanding capabilities. Notably, GroundingGPT surpasses the powerful baseline of Valley by 2.4%, 6.9% and 1.8% on MSVD-QA, MSRVTT-QA and ActivityNet-QA, respectively.

Object Hallucination We conduct an evaluation of MLLMs regarding their object hallucination in Table 6. GroundingGPT achieves the highest performance across all three sampling subsets. Notably, GroundingGPT-7B outperforms larger models such as InstructBLIP-13B, on the challenging Adversarial subset, GroundingGPT exhibits 14.07% increase in accuracy and 8.18% increase in f1-score, while experiencing a 27.60% decrease in the "Yes" metric. Additionally, on the "unanswerable" subset of VisWiz benchmark, GroundingGPT significantly improves from 67.8% to 84.0% compared to LLaVA-1.5. This improvement reveals that our model not only acquires a high-level semantic understanding of the overall image but also develops the ability to comprehend local details. This comprehensive understanding of the image enables the model to effectively suppress the occurrence of hallucinations.

4.2.3 Qualitative Results

We provide qualitative results to demonstrate the multi-modal understanding and grounding capabilities of our model. Figure 3 showcases examples illustrating the performance of GroundingGPT in multi-modal grounding tasks. More examples are available in appendix E. These results indicate that GroundingGPT excels in fine-grained multi-modal grounding tasks while maintaining a comprehensive understanding of multi-modal inputs.

4.3 Ablation Study

To validate the effectiveness of our approach, we conducted experiments on the REC task to assess the impact of training strategy, model architecture, and size on the results.

Training Strategy As shown in Table 7, it is evident that including fine-grained training data in the first stage results in a decline in performance.

S1	S2	S3	RefCOCO		
			val	testA	testB
C+F	F	C+F	82.43	86.87	75.37
C	F	C+F	**84.68**	**88.88**	**78.94**

Table 7: Ablation studies of the training strategy on the REC task. The S1 to S3, denoted as stage 1 to stage 3, represent the training data used in each stage. "C" represents coarse-grained data, while "F" represents fine-grained data. To quickly illustrate the performance, we adopt a simple training setting (224×224 simage resolution and linear adapter) as the toy model.

LLM Size	Adapter	RefCOCO		
		val	testA	testB
7B	Linear	86.01	90.45	80.43
7B	MLP	88.02	91.55	82.47
13B	MLP	**88.26**	**92.05**	**82.65**

Table 8: Ablation studies of the model architecture, LLM size on the REC task.

This can be attributed to the model's limited understanding of the images at this early stage. The introduction of fine-grained data during training may introduce interference and hinder the model's learning. This finding further validates the effectiveness of our coarse-to-fine training strategy.

Model Architecture and Size As shown in Table 8, the top two rows demonstrates that replacing the linear layer with an MLP in the adapter leads to performance enhancement. This improvement can be attributed to the preservation of more comprehensive image information and the improved mapping of image embeddings to the LLM space. Besides, increasing the LLM size leads to an improvement. This can be attributed to the fact that larger language model possess richer knowledge and stronger modeling capabilities.

5 Conclusion

In this paper, we introduce GroundingGPT, a unified end-to-end multi-modal grounding model. To the best of our knowledge, this is the first multi-modal large language model capable of performing multi-modal grounding and understanding tasks. We adopt a three-stage coarse-to-fine training strategy, accompanied by the construction of stage-specific training datasets, to effectively train the model. Our model demonstrates remarkable performance in multi-modal grounding and understanding tasks. Extensive experiments conducted on a wide range of MLLM benchmarks confirm the effectiveness and generality of our model. To foster further advancements in this field, we make our model, code, and dataset openly accessible.

6 Limitations

Sampling Strategy Due to computational memory constraints, GroundingGPT adopts a sampling approach when processing videos and audios. However, this method inevitably results in some loss of crucial information, especially when dealing with longer videos. One future research direction is to explore better modeling approaches for longer videos and minimize information loss.

Cross-modal Inputs At present, the majority of the training data primarily consists of single-modal inputs. However, further exploration is needed to address the challenges posed by multi-modal inputs. In the future, we plan to investigate methods for accomplishing grounding tasks in the context of simultaneous multi-modal inputs. For instance, we aim to simultaneously perform spatial and temporal grounding on input videos. Additionally, we will annotate such data to foster advancements in this field.

Grounding Ability Despite achieving promising results in multi-modal grounding tasks, GroundingGPT currently lacks the capability to output more fine-grained grounding results such as segmentation masks. In future work, we plan to expand the grounding tasks to support a broader range of grounding requirements.

References

Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. 2017. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, pages 5803–5812.

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

David Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In

Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies, pages 190–200.

Feilong Chen, Minglun Han, Haozhi Zhao, Qingyang Zhang, Jing Shi, Shuang Xu, and Bo Xu. 2023a. X-llm: Bootstrapping advanced large language models by treating multi-modalities as foreign languages. *arXiv preprint arXiv:2305.04160*.

Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. 2021. Localizing visual sounds the hard way. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16867–16876.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. 2023b. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*.

Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen. 2020. Clotho: An audio captioning dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 736–740. IEEE.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. 2023. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. 2017. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE international conference on computer vision*, pages 5267–5275.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. 2023. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617.

Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. 2023. Vtimellm: Empower llm to grasp video moments. *arXiv preprint arXiv:2311.18445*.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. 2021. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1780–1790.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017a. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pages 706–715.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017b. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73.

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

Kunchang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. 2023b. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023c. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi

Wang, Conghui He, Ziwei Liu, et al. 2023b. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Ruipu Luo, Ziwang Zhao, Min Yang, Junwei Dong, Minghui Qiu, Pengcheng Lu, Tao Wang, and Zhongyu Wei. 2023. Valley: Video assistant with large language model enhanced ability. *arXiv preprint arXiv:2306.07207*.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. 2023. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20.

Xinhao Mei, Chutong Meng, Haohe Liu, Qiuqiang Kong, Tom Ko, Chengqi Zhao, Mark D Plumbley, Yuexian Zou, and Wenwu Wang. 2023. Wavcaps: A chatgpt-assisted weakly-labelled audio captioning dataset for audio-language multimodal research. *arXiv preprint arXiv:2303.17395*.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. 2023. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*.

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649.

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

Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Erix Xing, Ming-Hsuan Yang, and Fahad S Khan. 2023. Glamm: Pixel grounding large multimodal model. *arXiv preprint arXiv:2311.03356*.

Yu Shu, Siwei Dong, Guangyao Chen, Wenhao Huang, Ruihua Zhang, Daochen Shi, Qiqi Xiang, and Yemin Shi. 2023. Llasm: Large language and speech model. *arXiv preprint arXiv:2308.15930*.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326.

Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. 2023. Pandagpt: One model to instruction-follow them all. *arXiv preprint arXiv:2305.16355*.

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

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296.

Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Faisal Ahmed, Zicheng Liu, Yumao Lu, and Lijuan Wang. 2022. Unitab: Unifying text and box outputs for grounded vision-language modeling. In *European Conference on Computer Vision*, pages 521–539. Springer.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, Chaoya Jiang, Chenliang Li, Yuanhong Xu, Hehong Chen, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. 2023. mplug-owl: Modularization empowers large language models with multimodality.

Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. 2023. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. 2019. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9127–9134.

Abhay Zala, Jaemin Cho, Satwik Kottur, Xilun Chen, Barlas Oguz, Yashar Mehdad, and Mohit Bansal. 2023. Hierarchical video-moment retrieval and step-captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23056–23065.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6720–6731.

Ao Zhang, Liming Zhao, Chen-Wei Xie, Yun Zheng, Wei Ji, and Tat-Seng Chua. 2023a. Next-chat: An lmm for chat, detection and segmentation. *arXiv preprint arXiv:2311.04498*.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023b. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*.

Hang Zhang, Xin Li, and Lidong Bing. 2023c. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*.

Hao Zhang, Hongyang Li, Feng Li, Tianhe Ren, Xueyan Zou, Shilong Liu, Shijia Huang, Jianfeng Gao, Lei Zhang, Chunyuan Li, et al. 2023d. Llava-grounding: Grounded visual chat with large multimodal models. *arXiv preprint arXiv:2312.02949*.

Yang Zhao, Zhijie Lin, Daquan Zhou, Zilong Huang, Jiashi Feng, and Bingyi Kang. 2023. Bubogpt: Enabling visual grounding in multi-modal llms. *arXiv preprint arXiv:2307.08581*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Implementation details

We provide more details of our experiment configuration for reproducing our model. We provide hyper-parameters for all stages in Table 9.

Settings	Stage1	Stage 2	Stage3
batch size	64	16	8
learning rate	1e-3	2e-5	2e-5
learning schedule	Cosine decay		
warm up ratio	0.03	0.03	0.03
weight decay	0.0	0.0	0.0
epoch	1	1	1
bf16	✓	✓	✓
tf32	✓	✓	✓
grad accumulate	1	2	2
DeepSpeed stage	ZeRO2		
GPUs	8×A100		

Table 9: The hyper-parameters for model training.

B Training Dataset Details

In Table 10, we provide a comprehensive list of the datasets used in constructing our training dataset. This includes the data utilized in all three stages. It should be noted that a significant portion of the data needs to be constructed in the desired format using publicly available data. Please refer to the section 3.2 for specific guidance on this matter. Additionally, we provide the statistical results of the training data we constructed in Table 11, including the number of samples for each stage, whether it is multi-turn dialogue data, average number of dialogue turns, average video duration, and average audio duration.

C Dataset Construction Templates

Table 12 presents the templates utilized for various tasks during the first two training stages. For the sake of demonstration, we provide three examples of instructions for each task.

D Fine-grained Instruction-tuning Dataset Generation Prompts

As shown in section 3.2.3, we use GPT-3.5 to generate the instruction-tuning dataset. For the image modality, in Figure 4, we provie the prompt we used to generate the detailed description dataset. In Figure 5, we provie the prompt we used to generate the conversation dataset. For the video modality, we provie the prompt we used to generate the video grounding instruction-tuning dataset in Figure 6.

E More Visualization

To demonstrate the performance of GroundingGPT in multi-modal grounding and understanding tasks, we present more visualizations in this section. Figure 7 and Figure 8 showcase the capability of the GroundingGPT model in multi-modal grounding tasks. Figure 9, Figure 10 and Figure 11 present the capability of GroundingGPT model in multi-modal understanding tasks.

Training Stage	Modality	Dataset source
Stage1	Image	LLaVA-Pretrain-595k
	Video	Valley-Pretrain-703k
	Audio	Wavcaps
Stage2	Image	RefCOCO, RefCOCOg, RefCOCO+, Visual Genome
	Video	DiDeMo, Charades-STA
	Audio	VGGSS
Stage3	Image	LLaVA-1.5-mix665k, Flickr30k Entities, VCR
	Video	Valley-Instruct-73k, Videochat-Instruct-11k, Activitynet Captions
	Audio	Clotho

Table 10: The publicly available dataset sources used for constructing the training data.

Training Stage	Modality	Samples	Multi-turn	Dialog Turns	Video Duration	Audio Duration
Stage1	Image	595K	✗	1	-	-
	Video	703K	✗	1	67.59s	-
	Audio	403K	✗	1	-	18.22s
Stage2	Image	4.2M	✗	1	-	-
	Video	80K	✗	1	38.03s	-
	Audio	5K	✗	1	-	9.97s
Stage3	Image	925K	✓	4.15	-	-
	Video	83K	✓	2.91	29.45s	-
	Audio	4K	✗	1	-	22.44s

Table 11: The statistics of the training data for the model, including the number of samples, whether it is multi-turn dialogue data, average number of dialogue turns, average video duration, and average audio duration.

Task	Template examples
Image Captioning	Provide a brief description of the given image. Write a terse but informative summary of the picture. Share a concise interpretation of the image provided.
REG	What object is present within the specified region<region>? Can you identify the item within the region<region>? Describe the object located within the region<region>.
REC	In this image, where is <exp> located? Can you identify the position of <exp> within this image? Please describe the location of <exp> in this image.
Object Attribute	What color is this <exp>? How many <exp> are visible within this image? How mang <exp> are there in the image?
Video Captioning	Relay a brief, clear account of the video shown. Offer a succinct explanation of the footage presented. Present a compact description of the clip's key features.
Video Dense Captioning	Describe the content shown in the video clip<time> of this video. What can you tell me about the video segment<time> in this video? Can you provide a description of the video snippet<time>?
Temporal Grounding	When did <event> occur in the video? Tell me the timestamp when <event> happened. At what time does <event> take place in the video?
Audio Captioning	Analyze the audio and provide a description of its content. Examine the audio and describe the different sounds present. Provide a detailed summary of the auditory elements in the audio clip.
Sound Localization	What is the cause of the sound in this given image? Can you pinpoint the source of the sound in this image? Describe the location of the sound's origin in this image.

Table 12: Instruction templates used to construct the training dataset in the first two stages. The templates include several placeholders: '<region>' represents the coordinates of a region in an image, '<exp>' represents the expression correspond to an image region, '<time>' represents a time segment in a video, and '<event>' represents an event to be located in a video. During the dataset construction process, these placeholders are replaced with corresponding information.

System Message

You are an AI visual assistant that can analyze a single image. You receive several sentences, each describing the same image you are observing. In addition, specific object locations within the image are given, along with detailed coordinates. These coordinates are in the form of bounding boxes,represented as [x1, y1, x2, y2], with floating numbers ranging from 0 to 1. These values correspond to the top left x, top left y, bottom right x, and bottom right y.

The task is to create an accurate description related to the image based on the information. The description should involve mentioning the position of objects in the image. The position can be represented in the format [x1, y1, x2, y2]. Description should be accurate and concise, limited to 100 words.

Here are some additional requirements about generated descriptions:
1. In description, you need to mention bounding box coordinates to refer to some objects or regions, instead of directly say the object name or describing the regions in text.
2. Avoid introducing objects that do not exist in the original descriptions and avoid including excessive subjective perceptions to prevent creating illusions.
3. Only describe what you are certain about, and avoid providing descriptions that may be ambiguous or inaccurate.
4. The boxes provided in different sentences may have some coordinates that are the same or very close, which could be because different expressions refer to the same object. You should analyze and avoid describing a single object within a bounding box as multiple distinct entities.
5. The number of words in the output description should not exceed 100 words. Keep it as concise as possible. You do not need to include everything in several descriptions. Choose the output you think is the most important.

In-context Example

User:

{'sentence': 'Two young guys with shaggy hair look at their hands while hanging out in the yard .',
'boxes': {'Two young guys': [[0.474, 0.248, 0.655, 0.668], [0.589, 0.218, 0.781, 0.744]], 'shaggy hair': [[0.538, 0.248, 0.616, 0.31], [0.592, 0.226, 0.718, 0.29]], 'their hands': [[0.517, 0.366, 0.592, 0.404], [0.471, 0.394, 0.571, 0.448]]}}

{'sentence': 'Two young , White males are outside near many bushes .',
'boxes': {'Two young , White males': [[0.474, 0.248, 0.655, 0.668], [0.589, 0.218, 0.781, 0.744]], 'many bushes': [[0.0, 0.438, 0.631, 0.944], [0.826, 0.428, 0.994, 0.672]]}}

{'sentence': 'Two men in green shirts are standing in a yard .',
'boxes': {'Two men': [[0.474, 0.248, 0.655, 0.668], [0.589, 0.218, 0.781, 0.744]], 'green shirts': [[0.619, 0.286, 0.769, 0.486], [0.517, 0.31, 0.649, 0.47]]}}

{'sentence': 'A man in a blue shirt standing in a garden .',
'boxes': {'A man': [[0.589, 0.218, 0.781, 0.744]], 'a blue shirt': [[0.619, 0.286, 0.769, 0.486]]}}

{'sentence': 'Two friends enjoy time spent together .',
'boxes': {'Two friends': [[0.474, 0.248, 0.655, 0.668], [0.589, 0.218, 0.781, 0.744]]}}

Assistant:

In the image, two young men[0.474, 0.248, 0.655, 0.668][0.589, 0.218, 0.781, 0.744] with shaggy hair[0.538, 0.248, 0.616, 0.310][0.592, 0.226, 0.718, 0.290] are seen in a yard surrounded by bushes[0.000, 0.438, 0.631, 0.944][0.826, 0.428, 0.994, 0.672]. They are focused on their hands[0.517, 0.366, 0.592, 0.404][0.471, 0.394, 0.571, 0.448]. It seems that they enjoy time spent together.

Figure 4: The system message and in-context example used for generating detailed description dataset.

System Message

You are an AI visual assistant that can analyze a single image. You receive several sentences, each describing the same image you are observing. In addition, specific object locations within the image are given, along with detailed coordinates. These coordinates are in the form of bounding boxes,represented as [x1, y1, x2, y2], with floating numbers ranging from 0 to 1. These values correspond to the top left x, top left y, bottom right x, and bottom right y.

The task is to create question-answer pairs related to the image based on the information. The conversation should involve mentioning the position of objects in the image. The position can be represented in the format [x1, y1, x2, y2]. The questions and answers should be limited to 50 words.

Here are some additional requirements about generated conversations:
1. In conversation , you need to mention bounding box coordinates to refer to some objects or regions, instead of directly say the object name or describing the regions in text.
2. Avoid introducing objects that do not exist in the original descriptions and refrain from including excessive subjective perceptions to prevent creating illusions.
3. Only describe what you are certain about, and avoid providing descriptions that may be ambiguous or inaccurate.
4. The boxes provided in different sentences may have some coordinates that are the same or very close, which could be because different expressions refer to the same object. You should analyze and avoid describing a single object within a bounding box as multiple distinct entities.
5. The questions and answers should be limited to 50 words. Keep it as concise as possible.

In-context Example

User:

{'sentence': 'A child in a pink dress is climbing up a set of stairs in an entry way .',
'boxes': {'A child': [[0.019, 0.302, 0.285, 0.678]], 'a pink dress': [[0.019, 0.38, 0.227, 0.596]], 'a set of stairs': [[0.0, 0.636, 0.304, 0.958]], 'an entry way': [[0.099, 0.052, 0.288, 0.614]]}}

{'sentence': 'A little girl in a pink dress going into a wooden cabin .',
'boxes': {'A little girl': [[0.019, 0.302, 0.285, 0.678]], 'a pink dress': [[0.019, 0.38, 0.227, 0.596]]}}

{'sentence': 'A little girl climbing the stairs to her playhouse .',
'boxes': {'A little girl': [[0.019, 0.302, 0.285, 0.678]], 'the stairs': [[0.0, 0.636, 0.304, 0.958]], 'her playhouse': [[0.069, 0.014, 0.997, 0.774]]}}

{'sentence': 'A little girl climbing into a wooden playhouse .',
'boxes': {'A little girl': [[0.019, 0.302, 0.285, 0.678]], 'a wooden playhouse': [[0.069, 0.014, 0.997, 0.774]]}}

{'sentence': 'A girl going into a wooden building .',
'boxes': {'A girl': [[0.019, 0.302, 0.285, 0.678]]}}

Assistant:

{'question':'What clothes is the girl[0.019, 0.302, 0.285, 0.678] wearing?',
'answer':'The girl is wearing a pink dress[0.019, 0.380, 0.227, 0.596].'}

{'question':'Where is the little girl[0.019, 0.302, 0.285, 0.678] going?',
'answer':'She is climbing into a wooden playhouse[0.069, 0.014, 0.997, 0.774].'}

{'question':'What is the child[0.019, 0.302, 0.285, 0.678] doing?',
'answer':'She is climbing up a set of stairs[0.000, 0.636, 0.304, 0.958].'}

{'question':'Why is this girl climbing the stairs?',
'answer':'She is going to enter hey playhouse[0.069, 0.014, 0.997, 0.774].'}

Figure 5: The system message and in-context example used for generating conversation dataset.

System Message
You are an AI visual assistant that can analyze a single video. You will receive descriptions of several segments of this video along with corresponding timestamps for each description, indicating the time range of the corresponding video segment. These timestamps are in the form of bounding boxes, represented as {t1,t2} with floating numbers ranging from 0 to 1. These values correspond to the start time of the video segment is represented by t1, and the end time is represented by t2. Design a conversation between you and a person asking about this video. Ask three to ten questions and give corresponding answers. The answers should be in a tone that a visual AI assistant is seeing the video and answering the question.

Here are some additional requirements about generated questions and answers:
1. Only include questions that have definite answers.
2. Always answer as if you are directly looking at the video.
3. Make the question as diverse as possible. Include questions asking about the visual content of the video, including the events within the video segments, the order in which different events occur, and other related details.
4. Avoid introducing objects that do not exist in the original descriptions and refrain from including excessive subjective perceptions to prevent creating illusions.
5. The number of words in the question and answer should not exceed 50.

In-context Example
User:
{'A young woman is seen standing in a room and leads into her dancing.': '{0.01, 0.24}'}

{'The girl dances around the room while the camera captures her movements.': '{0.21, 0.74}'}

{'She continues dancing around the room and ends by laying on the floor.': '{0.68, 0.96}'}
Assistant:
{'question':'What happens during the time period{0.01,0.24}?',
'answer':'A young woman is seen standing in a room and leads into her dancing.'}

{'question':'When does the women start dancing around the room?',
'answer':'The girl starts dancing around the room in {0.21,0.74}'}

{'question':'How does the girl end her dance routine?',
'answer':'The girl ends her dance routine by laying on the floor, which occurs in {0.68,0.96}.'}

Figure 6: The system message and in-context example used for generating video grounding dataset.

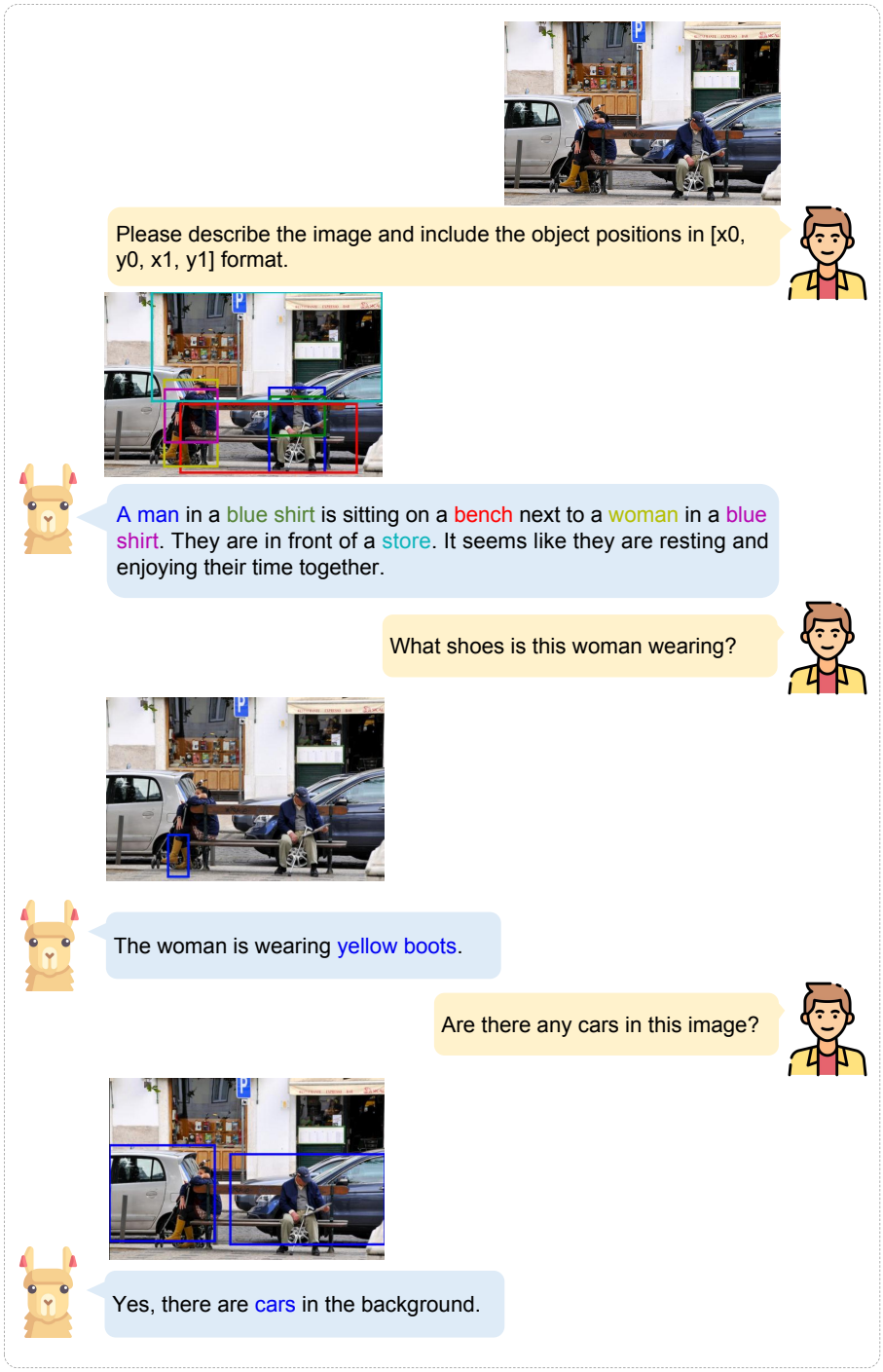


Figure 7: Examples of GroundingGPT's grounding capability in image modality.

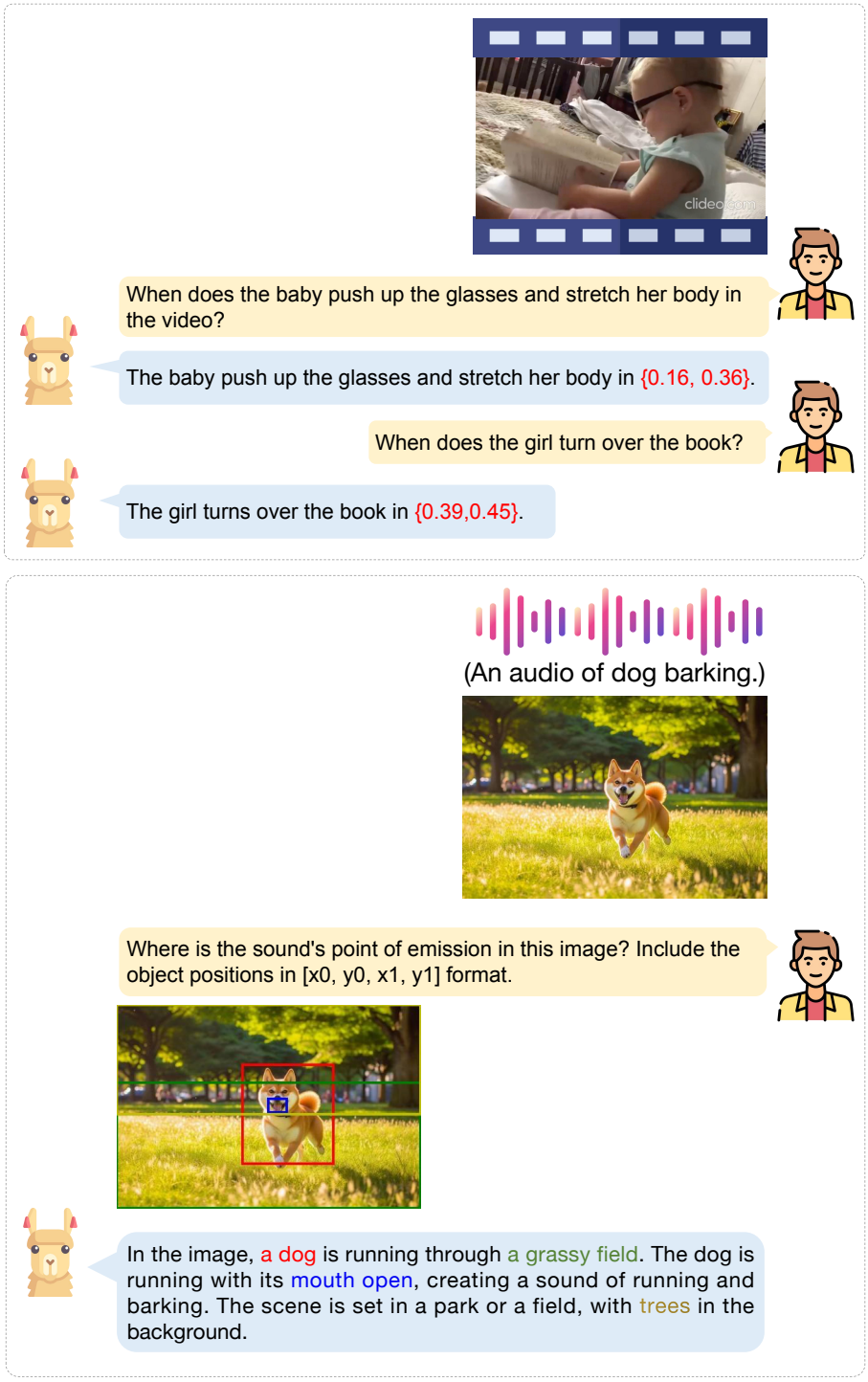


Figure 8: Examples of GroundingGPT's grounding capability in video and audio modality.

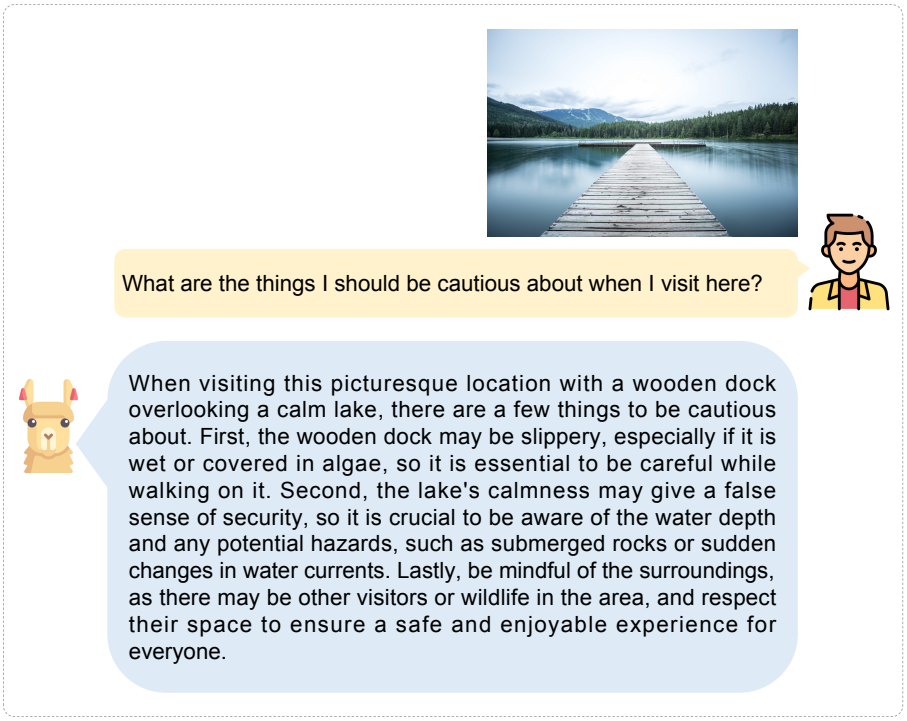


Figure 9: Examples of GroundingGPT's understanding capability in image modality.

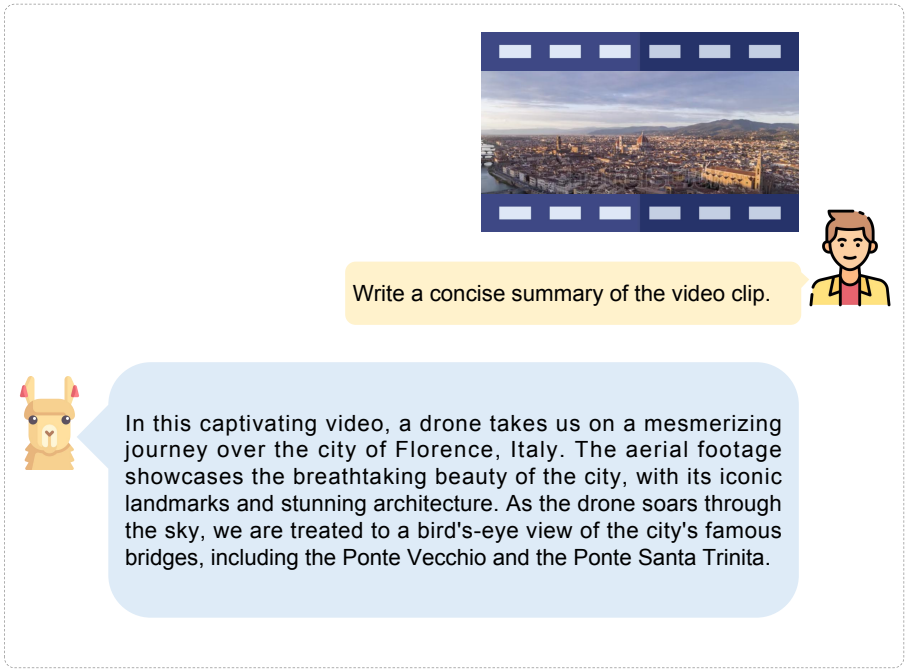
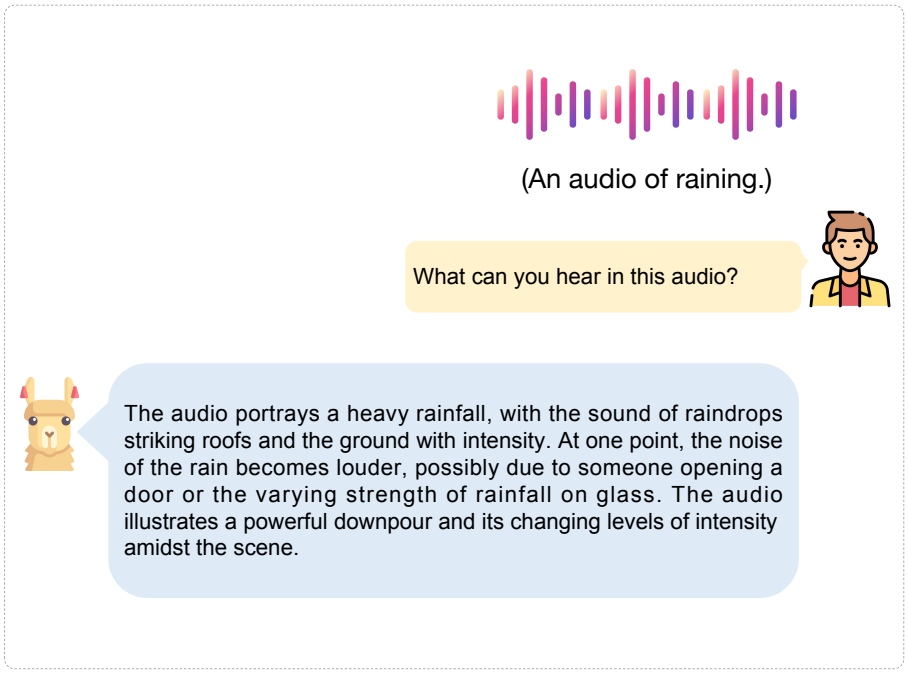


Figure 10: Examples of GroundingGPT's understanding capability in video modality.



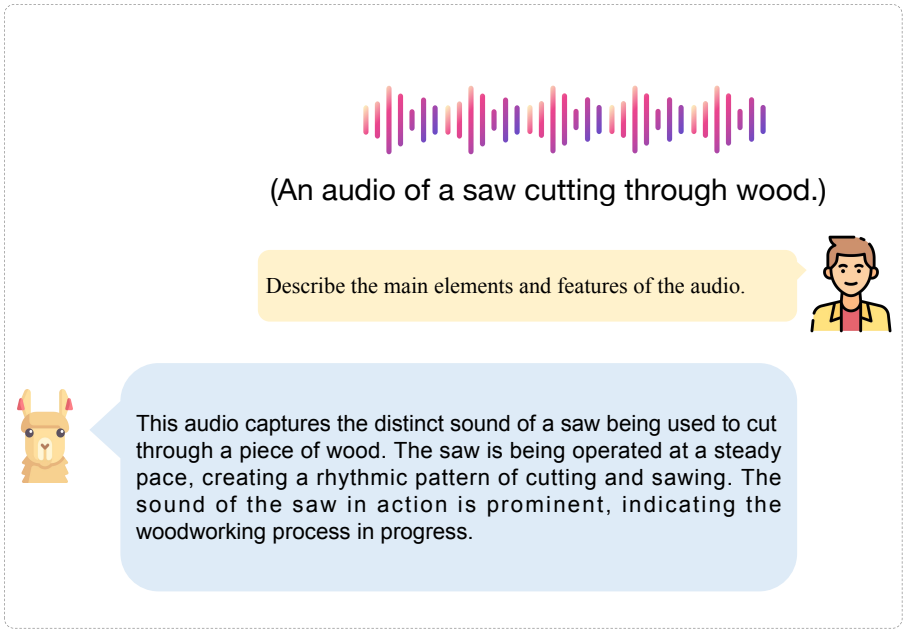


Figure 11: Examples of GroundingGPT's understanding capability in audio modality.