ChatSpot: Bootstrapping Multimodal LLMs via Precise Referring Instruction Tuning

Liang Zhao[1] , **En Yu**[2] , **Zheng Ge**[1] , **Jinrong Yang**[2] , **Haoran Wei**[1] , **Hongyu Zhou**[1] , **Jianjian Sun**[1] , **Yuang Peng**[3] , **Runpei Dong**[4] , **Chunrui Han**[1] , **Xiangyu Zhang**[1]
[1]MEGVII Technology
[2]Huazhong University of Science and Technology
[3]Tsinghua University
[4]Xian Jiaotong University
{zhaoliang06, gezheng}@megvii.com, yuen@hust.edu.cn

Abstract

Human-AI interactivity is a critical aspect that reflects the usability of Multimodal Large Language Models (MLLMs). However, existing end-to-end MLLMs only allow users to interact with them through language instructions, leading to the limitation of the interactive accuracy and efficiency. In this study, we present *precise referring instructions* that utilize diverse reference representations such as points and boxes as referring prompts to refer to the special region. This enables MLLMs to focus on the region of interest and achieve finer-grained interaction. Based on *precise referring instruction*, we propose ***ChatSpot***, a unified end-to-end MLLM that supports diverse forms of interactivity including mouse clicks, drag-and-drop, and drawing boxes, which provides a more flexible and seamless interactive experience. We also construct a multi-grained vision-language instruction-following dataset based on existing datasets and GPT-4 generating. Furthermore, we design a series of evaluation tasks to assess the effectiveness of region recognition and interaction. Experimental results showcase ChatSpot's promising performance. Project page: https://github.com/Ahnsun/ChatSpot.

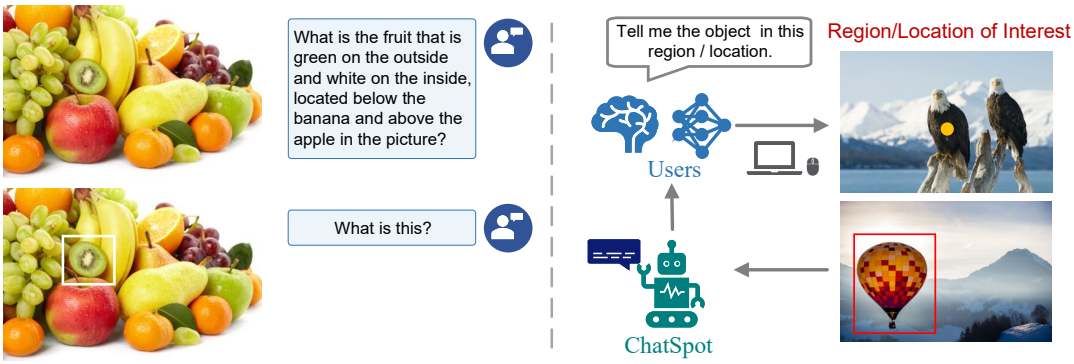
1 Introduction

Recent advances in large language models (LLMs) exemplified by GPT-3 [Brown *et al.*, 2020] and LLaMA [Touvron *et al.*, 2023] have demonstrated significant potential in the domain of zero-shot learning and logical reasoning. By aligning pre-trained LLMs to follow human language instructions through Reinforcement Learning with Human Feedback (RLHF) [Christiano *et al.*, 2017], InstructGPT [Ouyang *et al.*, 2022] and ChatGPT [OpenAI, 2023a] have showcased powerful capabilities for human-AI interaction, leading to a new paradigm shift towards the realization of artificial general intelligence (AGI).

Inspired by the remarkable success of GPT series [Brown *et al.*, 2020; OpenAI, 2023a; OpenAI, 2023b], researchers attempt to incorporate more modalities into LLMs for multimodal human-AI interaction, with vision-language interaction being an important topic of focus. In order to incorporate visual modality into LLM, significant processes have been made to bridge the gap between LLMs and vision foundation models. There are two mainstream paradigms for building multimodal large language models (MLLMs). One is *plugin-based* MLLM [Wu *et al.*, 2023; Yang *et al.*, 2023b; Shen *et al.*, 2023] that utilizes off-the-shell LLMs [OpenAI, 2023a; Touvron *et al.*, 2023; Chiang *et al.*, 2023] as central controllers to schedule different visual expert models as plugins. In this way, the users can interact with LLMs to achieve diverse visual functions. Another paradigm is *end-to-end* MLLM [Alayrac *et al.*, 2022; Huang *et al.*, 2023; Liu *et al.*, 2023b] that employs various techniques to align visual signals obtained from the vision encoder to the language semantic space and input vision tokens and language tokens together into the large language decoder.

Despite existing *end-to-end* MLLMs have achieved remarkable progress in vision-language human-AI interaction, the mode of interactive instruction is still limited in natural language. When meeting complex scenes as illustrated in Fig. 1 (a), it is difficult to only use the language to accurately describe the requirement of the user. However, if we can add some **referring prompts**, *e.g.*, reference points, bounding boxes, *etc*, to MLLMs, the model can focus on the region of interest (RoI) and achieve finer-grained interaction, which is more flexible and user-friendly.

Motivated by this, we present ***ChatSpot***, a fully end-to-end unified multimodal language model designed to empower special region vision-language interaction. As illustrated in Fig. 1 (b), ChatSpot extends the LLMs' power to incorporate diverse multimodal inputs, and it can support a range of interaction forms. Users can communicate with the system using their native language, as well as gestures such as clicking and drawing boxes that we call ***Precise Referring***, to obtain the desired information about the entire image or the region of interest (RoI). When a specific region is selected, ChatSpot can follow the precise referring instructions to perform various fine-grained applications, such as identifying jersey numbers or analyzing facial expressions in the given task, which is illustrated in Fig. 4. Furthermore, the precise referring can be



(a) Language **vs.** Referring prompt + Language (b) ChatSpot overview

Figure 1: (a) is the intuitive comparison between language instruction and the combination of region prompt and language instruction. (b) is the overview of ChatSpot. We extend the power of advanced LLM to vision-language modality and support a range of interaction forms including native language, mouse-clicking, and mouse boxing, enabling the interaction to be more flexible and user-friendly.

regarded as a link of the chain-of-thought (CoT) to enhance the special logical reasoning ability of MLLMs. When an intelligent agent (robot or expert model) locates a target or region of interest based on user demands, ChatSpot can further analyze the details of this region and provide more specific suggestions and refined instructions, enabling the agent to interact with the physical world more effectively and precisely.

The success of ChatSpot hinges on **three** components:
(1) We design a simple but effective *precise referring instruction* tuning method for MLLMs to support fine-grained and flexible human-AI interaction.
(2) We construct a high-quality Multi-Grained Vision-Language Instruction-following Dataset (**MGVLID**) including image-level and region-level multimodal SFT data with around $1.2M$ images and $3M$ query-answer pairs by collecting from existing datasets and generated based on GPT-4.
(3) We design a series of evaluation tasks and metrics to assess the effectiveness of the proposed MLLM.

Extensive experiments have been conducted on a wide of vision-centric and vision-language benchmarks, and our ChatSpot shows excellent performance.

2 Related Works

2.1 Large Language Models

In recent years, large language models (LLMs) have garnered considerable attention in the domains of natural language processing (NLP) and artificial general intelligence (AGI) owing to their remarkable performance in language generation, in-context learning, world knowledge, and logical reasoning. Early works, *e.g.*, BERT [Devlin *et al.*, 2018], GPT-2 [Radford *et al.*, 2019] and T5 [Raffel *et al.*, 2020] established the foundation architecture of LLMs. Then, with the release of GPT-3 [Brown *et al.*, 2020], the first-ever language model to reach the parameter size of 175 billion, LLM achieved impressive zero-shot performance on various language benchmarks. Furthermore, researchers discovered *emergent ability* [Wei *et al.*, 2022] in LLMs. That is when the model size of language models scales up to a certain level, there is a qualitative leap in the capabilities of language models. Sequentially, by aligning pre-trained GPT-3 to follow human instructions through Reinforcement Learning with Human Feedback (RLHF) [Christiano *et al.*, 2017], InstructGPT [Ouyang *et al.*, 2022] and ChatGPT [OpenAI, 2023a] showcased powerful capabilities for human-AI interaction, which make LLMs reach its "iPhone moment". Inspired by the great success of GPT series, many other open-sourced LLMs, such as OPT [Zhang *et al.*, 2022b], LLaMA [Touvron *et al.*, 2023], and GLM [Zeng *et al.*, 2022], have been proposed, which achieve similar performance to GPT-3. Based on these open-sourced LLMs, several specific fine-tuned models are proposed to construct LLMs for various applications. For instance, Alpaca [Taori *et al.*, 2023] proposes a self-instruct framework based on LLaMA [Touvron *et al.*, 2023] and employs $52K$ instructions generated by ChatGPT [OpenAI, 2023a] to construct an exceptional dialogue model.

2.2 Human-AI Interactivity

Once we have sufficiently powerful LLMs, the next step is to figure out how to make these large models better meet human needs, with human-AI interaction being the core issue, that is, how to enable large models to more efficiently receive and understand human instructions. The powerful zero-shot and logical reasoning ability of LLM makes it the central controller of the interactive system to schedule various application tools for different modality tasks, such as VQA, image editing, and image captioning. There are two mainstream interactive styles, *i.e.*, plugin-based and end-to-end interaction. Plugin-based methods [Wu *et al.*, 2023; Yang *et al.*, 2023b; Shen *et al.*, 2023; Liang *et al.*, 2023; Yang *et al.*, 2023a] usually prompt LLM (ChatGPT [OpenAI, 2023a], GPT-4 [OpenAI, 2023b] or LLaMA [Touvron *et al.*, 2023]) to invoke different plugins from other foundation or expert models to perform specific functions according to human instructions. However, despite plugin-based methods that enable diverse applications, they are limited in the effectiveness of plugin invocation and the performance of the plugin model. On the contrary, end-to-end interactive systems usually use a single large multimodal model to accomplish interaction. This approach takes advantage of cross-modal transfer, aligning multimodal domains to a common language semantic space and then using autoregressive language models as decoders to output the language. Follow-

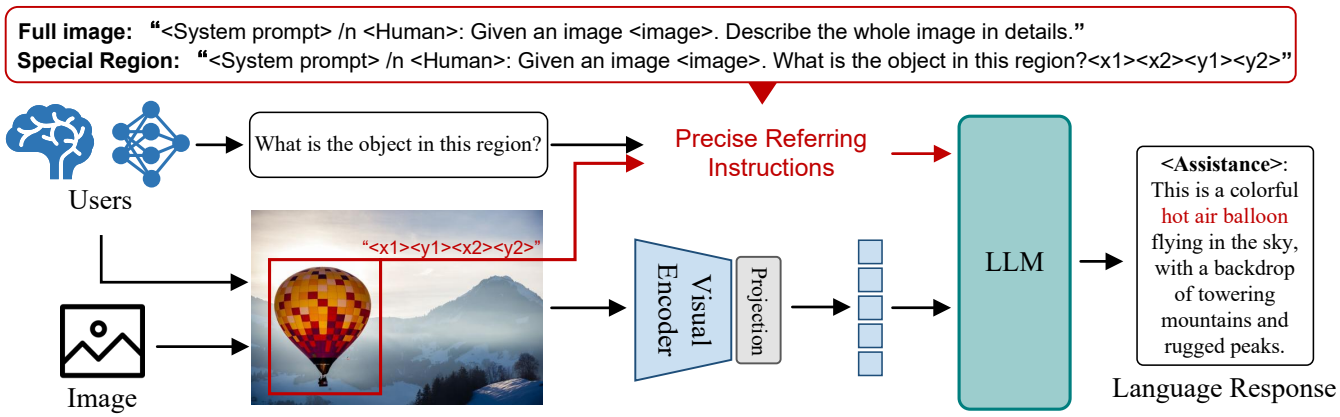


Figure 2: **Overall pipeline of ChatSpot**. The architecture of ChatSpot consists of three main components: (1) an image encoder, (2) a large language model, and (3) a modality-align projector.

ing this pipeline, Flamingo [Alayrac *et al.*, 2022] developed a gated cross-attention trained on billions of image-text pairs to align vision and language modality, which shows strong performance in few-shot learning. BLIP-2 [Li *et al.*, 2023] introduced Q-Former to align visual features with language space more effectively. More recently, LLaVA [Liu *et al.*, 2023b] and LLaVA-like methods [Yu *et al.*, 2023; Wei *et al.*, 2024; Dong *et al.*, 2023] proposed to use a linear layer to replace Q-Former and design a two-stage instruction-tuning procedure. Although existing end-to-end methods achieve remarkable performance in high efficiency, they are all limited to the interaction form of the full image and language-only instruction, which can not satisfy the demand for the specific region interaction. In this work, we build an end-to-end unified multimodal language model that supports a range of interaction forms that supports both full images and specific region.

3 Methods

3.1 Overall Architecture

As illustrated in Fig. 2, ChatSpot consists of an image encoder, and a decoder-only LLM, and a modality alignment block. Inspired by LLaVA [Liu *et al.*, 2023b], ChatSpot incorporates a simple multilayer perceptron (MLP) to align the visual tokens with the space of language. The overall architecture is simple and does not use any extra AI models or post-processing operations. Different from previous end-to-end interactive systems [Liu *et al.*, 2023b; Zhu *et al.*, 2023] that only support full image interaction, ChatSpot presents a more flexible interaction that supports users in further selecting the region of interest (RoI) to issue finer-grained instructions.

When an image I is uploaded by users to the ChatSpot system, users can use a mouse to select the RoI (points or boxes) R_t through a series of gestures, such as clicking and drawing boxes, and give some language instructions $X^{\texttt{instruct}}$ about these RoIs. The system then converts the position of R_t into the region prompt and connects it with the $X^{\texttt{instruct}}$ to generate the precise referring instructions. Afterward, the image I is inputted into the visual encoder to extract visual tokens. And then the modality-align projector transforms the visual tokens to the language semantic space. After obtaining the refined visual tokens and precise referring instructions. LLM decoder $\mathcal{F}$ takes them as inputs and generates the response language sequence Y autoregressively. Formally,

$$\begin{aligned} V &= \zeta \circ \mathcal{G}(I), \\ Y &= \mathcal{F}\Big(V, \Phi(R_t), X^{\texttt{instruct}}\Big), \end{aligned} \tag{1}$$

where V are aligned visual tokens. $\mathcal{G}$ is the visual encoder and ζ denotes the vision-language alignment projection. $\mathcal{F}$ is the large language decoder. $\Phi(\cdot)$ is the normalization operation.

3.2 Precise Referring Instruction

Due to the inherent semantic unit mismatch between images and texts, it is ineffective to directly use the whole image and language sentences to describe the vision-language task. To this end, we propose *precise referring instruction* that enables the unification of multi-grained vision-language task descriptions and supports proxy interaction forms. Specifically, we divide the instructions into two types, *i.e.*, image-level instructions and region-level instructions.

Image-level Instructions. The image-level instructions are usually used to describe the task of the whole image, and existing multimodal instructions mostly adopt this form. For instance, given an image and we want to know what the content of the image is. Then the instruction can be like *"Given an image* `<image>`*. Describe the whole image in detail"*, or *"Given an image* `<image>`*, please tell me:* `<question>`*"*, where `<image>` is the input image and `<question>` denote some relative questions about images. Afterward, the LLM ingests the whole sentence and outputs the response.

Region-level Instructions. Compared to the global information about the whole image, we often pay more attention to the information in specific regions. Therefore, it is valuable to design effective region-level instructions for MLLMs. The key challenge of region-level instructions is how to make LLM aware of the specific location of the region of interest. Here, we provide a simple but effective instruction format to achieve it. Specifically, we first define a unified region representation format as a tuple $R_t = \{x_k, y_k\}_{k=1}^{N}$ that represents N points located in the selected region. Then the coordinates

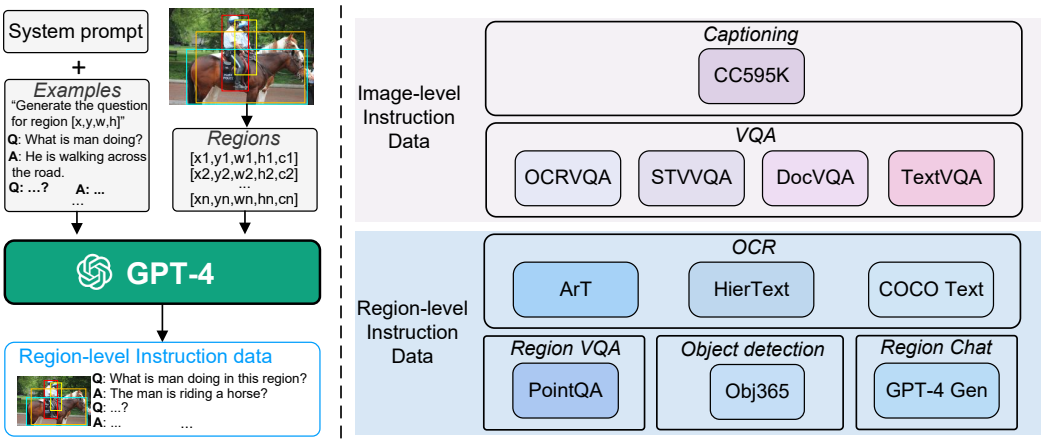


Figure 3: Illustration of the pipeline to collect region-level chatting data for MGVILD (left) and dataset groups included in Multigrained Vision-language Instruction Datasets, **MGVLID** (right).

of selected points are normalized to $[0, 1]$ and transferred to the text tokens as $\Phi(R_t)$. Finally, the region coordinate tokens are connected with the language instructions to generate the final region-level instructions. ([Liu *et al.*, 2023b] and [Bubeck *et al.*, 2023] have served as evidence that LLM possesses the capability to comprehend spatial relationships and coordinates based on textual descriptions.) A simple example of region-level instruction is as follows: *"Given an image* `<image>`*, What is the object doing in the region?* `<region>`*"* where `<region>` = `<box>`$\Phi(R_t)$`</box>`, `<box>` and `</box>` are special tokens to tell the LLM that this is a set of coordinates of the RoI. Notably, the number of selected points N is set freely so that we can achieve multi-grained interaction, such as points, boxes, and polygons.

3.3 Multi-grained Vision-language Instruction-following Dataset

In order to empower ChatSpot with the precise referring instruction following ability, we design a data collection pipeline with the assistance of GPT-4 [OpenAI, 2023b], as shown in Fig. 3. Inspired by LLaVA [Liu *et al.*, 2023b], we use the captions and bounding boxes of the target as the prompts and leverage GPT-4 to refine these captions and generate more relative and diverse conversation data. The difference is that our approach distinguishes itself by enforcing the alignment of every generated dialogue with precise regional coordinates. To achieve this, we leverage the VisualGenome dataset [Krishna *et al.*, 2017], which provides comprehensive annotations of objects, attributes, and relationships within each image, enabling us to construct region-level instruction following datasets. These datasets consist of dense region-wise captions organized alongside carefully curated seed examples, which are used to query GPT-4 in an in-context-learning fashion. Through this pipeline, we have successfully gathered a total of $108K$ region-level instruction following samples. Based on this data generation pipeline, we build a high-quality **M**ulti-**G**rained **V**ision-**L**anguage **I**nstruction-following **D**ataset, named **MGVLID**. MGVLID consists of two main parts, *i.e.*, image-text instruction-following data and region-text instruction-following data. The former data consists of the image and the caption of the entire image, while the latter consists of the image, the bounding boxes of the target in the image, and the target captions. As shown in Fig. 3, the whole MGVLID covers 11 source datasets and we hold out 4 datasets for model evaluation purposes.

Image-level Instruction Data. To gather image-level instruction-following data, we collect a wide range of publicly available multimodal datasets that have been human-annotated. We then transform these datasets into a unified instruction-following format. Specifically, we assemble a plethora of commonly used Question-Answering (QA), captioning, and object detection datasets, including CC595K (filtered based on CC3M [Sharma *et al.*, 2018]), OCR-VQA [Mishra *et al.*, 2019], ST-VQA [Biten *et al.*, 2022], DocVQA [Mathew *et al.*, 2021], TextVQA [Singh *et al.*, 2019] and Object365 [Shao *et al.*, 2019]. For each dataset, we design a series of unique instruction templates. These templates are subsequently carefully filtered and refined manually to ensure optimal rationales and diversity of the conversation. Due to the considerable differences in label lengths among various task datasets (such as caption or category words), we incorporate additional instruction tags to specify the desired response style. For instance, we include tags like *"answer in shot"* for short-answer data and *"answer in detail"* for long-answer data.

Region-level Instruction Data. While image-level instruction data empowers the model's global visual perception and human instruction-following ability, region-level instruction data offers region-level observation and more fine-grained instructions, enabling the model to further acquire spatial perception and reasoning abilities. In order to construct region-level instruction datasets, we first collect region-text pairs based on existing region-level task (object detection and OCR) datasets, *i.e.*, Object365 [Shao *et al.*, 2019], COCO text [Veit *et al.*, 2016], HierText [Long *et al.*, 2022] and Art [Chng *et al.*, 2019]. We collected region-text pairs that consist of instance-level bounding boxes and their corresponding content. Subsequently, we utilize unique instruction templates to further refine these region-text pairs, resulting in a series of questions and answers. Furthermore, we

Method	Backbone	Training Data	Region	COCO [Lin *et al.*, 2014]						
				AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l	Acc.
Multi-modal Large Language Models										
ChatSpot	CLIP-ViT	MGVLID	GT boxes	48.8	48.8	48.8	35.1	56.0	60.3	64.5
ChatSpot	CLIP-ViT	MGVLID	DINO boxes	39.6	50.2	44.1	21.6	45.8	58.8	-

Table 1: **Zero-shot region recognition results on COCO `val` set.** We randomly select $1,000$ images from the COCO validation set for evaluation. The referring regions are provided by GT boxes and advanced detector DINO.

VE	Training Data	OCR	VQA		
		COCO Text	VizWiz	PointQA (B)	PointQA (P)
CLIP	MGVLID	31.8	63.0	68.2	62.0

Table 2: **Experimental results on a diverse set of downstream tasks.** We also evaluate ChatSpot on a series of downstream tasks including optical character recognition (OCR) and visual question answering (VQA). We mainly report the metric of Accuracy (%) for evaluation. For PointQA, "B" and "P" mean that answer the question based on the given box and point, respectively.

Type	Method	Size	RefCOCO		
			val	**testA**	**testB**
Tra.	OFA-L [Wang *et al.*, 2022]	-	80.0	83.7	76.4
	TransVG [Deng *et al.*, 2021]	-	81.0	82.7	78.4
	VILLA [Gan *et al.*, 2020]	-	82.4	87.5	74.8
	UniTAB [Yang *et al.*, 2022]	-	86.3	88.8	80.6
LLM	VisionLLM-H [Wang *et al.*, 2023]	-	-	86.7	-
	Shikra-7B [Chen *et al.*, 2023a]	7B	87.0	90.6	80.2
	Shikra-13B	13B	87.8	91.1	81.7
	Qwen-VL-chat [Bai *et al.*, 2023]	7B	88.6	92.3	84.5
	Next-chat [Zhang *et al.*, 2023]	7B	85.5	90.0	77.9
	ChatSpot (Ours)	7B	**88.1**	**91.4**	**83.9**

Table 3: **Comparison with popular methods on RefCOCO**. Benefiting from the precise referring instruction tuning, ChatSpot can achieve 88.1% accuracy on RefCOCO val, which is on par with the 7B Qwen-VL-chat that employed a significantly larger dataset in comparison to that utilized by ChatSpot. Tra.: Traditional model.

also collect the PointQA datasets from LookTwice-QA [Mani *et al.*, 2020] to support point-wise referring instruction tuning, where the models are asked to answer questions based on the input points or boxes. By incorporating the high-quality dense region chatting data generated based on GPT-4, the final region-level instruction data is constructed.

4 Experiments

4.1 Implementation Details

To build ChatSpot, we implement the CLIP ViT-L/14 [Radford *et al.*, 2021] as the visual encoder to encode images. For the large language model, we choose open-sourced Vicuna-7B [Chiang *et al.*, 2023] as the language decoder, a LLaMA model fine-tuned with instructions. For alignment projection, we just adopt a simple linear layer to connect vision and language embedding space.

Inspired by LLaVA, the model is trained in a two-stage fashion. Firstly, we initialize the model using pre-trained weights from LLaMA and CLIP ViT. During this first stage, we only train the projection layer. Meanwhile, we freeze the majority of the LLM parameters. In this stage, we mainly use the image-text instruction-following data of MGVLID to train the model for vision-language instruction-following alignment learning. In the second stage, we only freeze the visual encoder and unfreeze the LLM parameters. In this stage, we mainly use the region-text instruction-following data including RegionChat to train the model for region-level instruction-following and multi-turn chatting ability. Specifically, the model is fine-tuned over 3 epochs, with a batch size of 128. AdamW [Loshchilov and Hutter, 2019] optimizer is employed, and the learning rate is set to $2e-3$ in the first training stage and $2e-5$ in the second training stage. For LLM, the maximum length of tokens is set to $2,048$.

4.2 Task Evaluation

In this part, we select several downstream tasks and general multimodal benchmarks to showcase ChatSpot's region recognization and zero-shot ability. Notably, all the experiments are conducted by the shared-parameter generalist model, and we just change the language instructions for different tasks.

Regional Classification. Object detection is a fundamental vision task that consists of object location and recognition subtasks. In this part, we mainly evaluate the regional classification ability of ChatSpot in COCO [Lin *et al.*, 2014], which is a common dataset in object detection tasks. Specifically, we first use the GT boxes or the bounding boxes generated by SOTA detectors, such as DINO [Zhang *et al.*, 2022a], as region prompts to ask ChatSpot to answer what category it is. For an example, we use *"What can you see in this region? `<region>`"*, where `<region>` denotes the coordinates of the region boxes. Then we compute the metrics of Average Precision (AP) and Accuracy about the bounding boxes with the predicted classes. Notably, due to ChatSpot's output typically being a single sentence, it cannot be directly used as a category for evaluation. Here, we employ the CLIP text encoder to calculate the text feature similarity between the output and all COCO categories for category determination.

As shown in Table 1, we randomly select $1,000$ images from the COCO validation set, namely COCO-1000, to evaluate ChatSpot for efficiency. Our ChatSpot achieves 64.5% accuracy on COCO-1000 with the provided GT boxes. When given DINO-generated bounding boxes as region prompts, our ChatSpot achieves 39.6% AP, which is also a competitive performance. Notably, We do not use any annotations from COCO. The results show that ChatSpot achieves impressive zero-shot classification ability in region-level recognization.

Regional Optical Character Recognition. Optical Character Recognition (OCR) is a visual entity recognition task that

Method	MMB_d	MMB_t	MM-Vet	POPE
BLIP-2 [Li *et al.*, 2023]	-	-	22.4	-
InstructBLIP [Dai *et al.*, 2024]	36.0	33.9	26.2	88.6
Shikra [Chen *et al.*, 2023b]	58.8	60.2	-	86.9
Qwen-VL† [Bai *et al.*, 2023]	38.2	32.2	-	84.7
Qwen-VL-Chat†	60.6	61.8	-	-
LLaVA-1.5 [Liu *et al.*, 2023a]	64.3	59.5	30.5	83.3
ChatSpot (Ours)	**64.5**	**63.6**	**37.5**	**91.4**

Table 4: **Comparison with SOTA methods on MLLM benchmarks.** We select MMBench and MM-Vet for general evaluation and choose POPE for the zero-shot object hallucination evaluation. † means the model is trained using additional in-house data.

requires the recognition of the graphemes in a written text. In this part, we select COCO text [Veit *et al.*, 2016] to evaluate the regional text recognization ability of ChatSpot. COCO Text is a large-scale dataset for text detection and recognition. We first use the provided region boxes of the datasets as the region referring and ask the ChatSpot *"What text can you see in this region?"* `<region>`. Then ChatSpot will respond to the specific answer. When the answered sentence includes the correct GT answer, we consider ChatSpot's response to be correct. As shown in Tab. 2, our ChatSpot achieves 31.8% accuracy on the COCO Text validation set.

Referring Expression Comprehension (REC). REC is also an important task for evaluating the regional referring ability of the model. We have additionally included experiments on the RefCOCO dataset and conducted a comprehensive comparison with existing SOTA solutions including traditional CV methods and LLM-based methods. As shown in Tab. 3, ChatSpot achieves promising performance compared to other generalist models, which shows the powerful spatial perception and referring instruction ability.

Visual Question Answering (VQA). VQA is the task of answering open-ended questions based on the whole image or the region of interest (RoI), which is well-suited for evaluating the perceptual and reasoning abilities of MLLMs in understanding image content. In this part, we choose two datasets to evaluate ChatSpot. One is VizWiz-VQA-Grounding dataset [Gurari *et al.*, 2018], a dataset that visually grounds answers to visual questions asked by people with visual impairments. Another is PointQA [Mani *et al.*, 2020], a set of datasets that require a pointer to an object in the image to be answered correctly. These two datasets both provide specific regions (boxes or points) and ask the question about the corresponding area. Therefore, it requires the model to possess the ability of region-level perception and reasoning ability. Specifically, given a region of interest and question, we first input the coordinate of region boxes to ChatSpot with the question. If the output answer sentence of ChatSpot includes the GT answer, we consider the response to be correct. As shown in Tab. 2, benefiting from ChatSpot's strong region-level instruction following abilities, the model achieves competitive performance that obtains 63.0% accuracy on VizWiz and 68.2% accuracy on PointQA (boxes given) and 62.0% on PointQA (points given).

General Multimodal Understanding. In order to showcase the general multi-modal ability of ChatSpot, we further evaluate ChatSpot on recent benchmarks proposed for evaluating the comphrehensive capabilities of MLLMs. As shown in Tab. 4, we select several mainstream MLLM benchmarks including MMBench, MMVet and POPE to evaluate ChatSpot. On MMBench, we report results on the both development and test sets. On POPE, we mainly report the results of "random" setting. For fair comparison, we utilizd the SFT data of LLaVA-1.5 (LLaVA665K). It is encouraging that ChatSpot outperforms the strong baseline LLaVA-1.5 across all three benchamrks. The experimental outcomes demonstrate the superior comprehensive capabilities of ChatSpot, which means powerful referring instruction ability is also beneficial to the general multi-modal understanding and reasoning.

4.3 Qualitative Analysis

In order to provide a comprehensive showcase of ChatSpot, we selected several classic cases to demonstrate its specific abilities. We specifically demonstrate four core capabilities of ChatSpot through these examples as follows:

Region Perception Ability. As shown in Fig. 4 (left top), we first show the case that depicts ChatSpot's ability to perceive the region of interest and recognize the corresponding context. In this case, ChatSpot can identify the selected area of different levels of granularity, *e.g.*, the head of the brown bear and the nose of the brown bear. It can also perceive some specific features of the target like the wet hair of the bear. By collaborating with human referring prompts (point or boxes), ChatSpot showcases its powerful capability in perceiving details, which provides sufficient detailed information for robots to perform more refined operations.

Content Generation Capability. ChatSpot also possesses a powerful content generation capability related to regions of interest, as illustrated in Fig. 4 (left-right). In this case, ChatSpot first recognizes the subject object (lemon) in the region of interest. Then it can also generate responses for other information that cannot be captured in the image, *i.e.*, the precautions when cutting a lemon with a knife. Thanks to the huge knowledge of LLMs, ChatSpot can provide rich content explanations or suggestions based on visual information.

Optical Character Recognition (OCR) Ability. ChatSpot also achieves impressive performance in recognizing the optical character, *e.g.*, text, number, and signal. As shown in Fig. 4 (bottom left), ChatSpot can accurately identify the number written on the signboard and analyze the purpose of these numbers based on the global context information.

Special Reasoning Ability. In addition to perception, another important capability of ChatSpot is spatial reasoning which can further analyze the region of interest based on its knowledge after recognition, which is important for robotics and automation. As shown in Fig. 4 (bottom right), ChatSpot first identifies the fridge and then determines that the refrigerator is powered on according to the power wine being plugged into the power strip and connected to the refrigerator. Furthermore, ChatSpot provides a series of detailed action instructions when the user wants to take a Coke from the fridge. This enables the robots to be able to make further decisions regarding fine-grained operations after inferring the specific status (fridge is plugged in.) of the RoI.

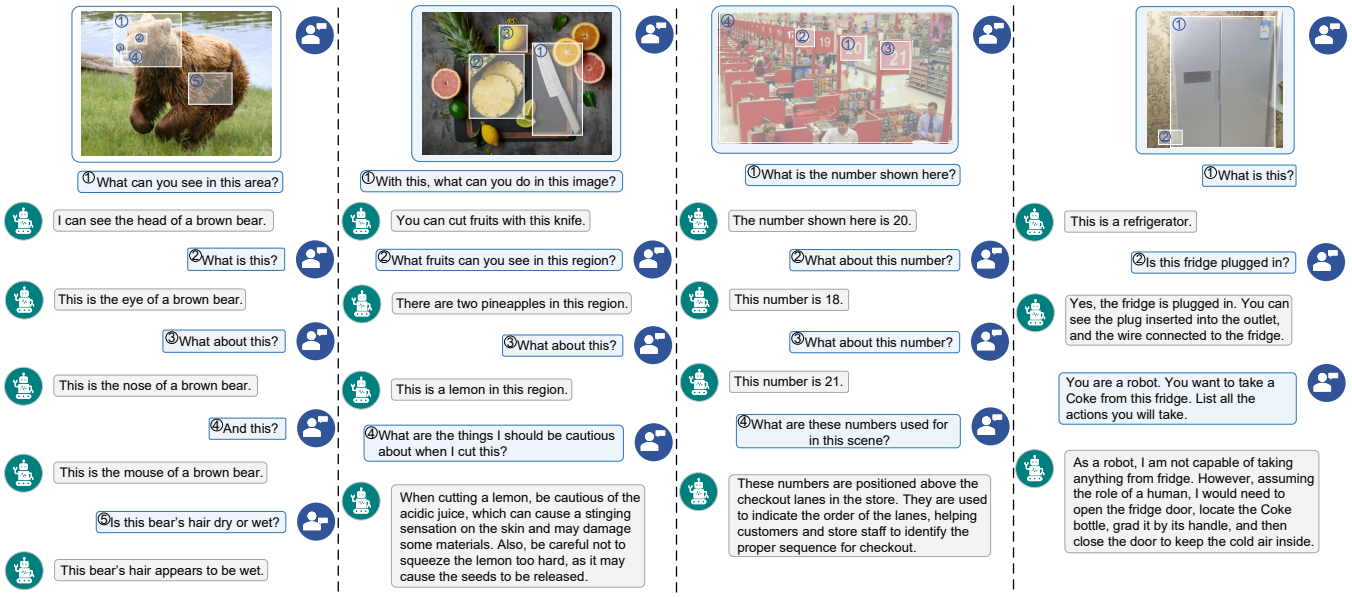


Figure 4: **Cases of interacting with ChatSpot** through drawing bounding boxes.

5 Discussion

In this section, we are dedicated to delving deeper into the capabilities and features of ChatSpot, as well as identifying the limitations that currently hinder further enhancement of ChatSpot's abilities.

5.1 Robustness on Region Referring

The process of selecting regions, whether through drawing boxes or clicking, plays a crucial role in ChatSpot. However, users often face difficulties in accurately annotating their areas of interest. In such instances, it is essential for ChatSpot to exhibit a high level of robustness in region selection. Hence, an analysis of the robustness of ChatSpot in the region referring is conducted. Specifically, we randomly introduce box noises of different scales (scale = 0.1, 0.2, and 0.3) to the region boxes of the COCO and VizWiz as illustrated in Fig. 5 (left). Fig. 5 (right) demonstrates that the performance of ChatSpot on COCO and VizWiz does not show a significant decrease after introducing box noises into the region bounding box, which means ChatSpot possesses strong robustness in region referring.

5.2 Limitations

Although ChatSpot has achieved remarkable performance in precise region referring and special region reasoning, it still has some noticeable limitations. For example, ChatSpot currently lacks the ability to support the recognition of certain special symbols, such as license plate numbers. These shortcomings can be attributed to insufficient training data. Furthermore, it is hard to evaluate the region recognition ability of multi-modal large language models since its evaluation is essentially different from traditional visual-language models. Though we take the first step to quantitatively evaluate it by giving pre-defined boxes, it is still an open problem: *how can we establish a comprehensive and automatic benchmark to evaluate existing multimodal large language models?* These limitations require further research in the future.

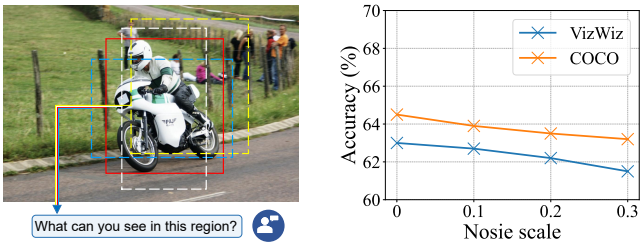


Figure 5: **Experiment on the robustness of region referring.** The left subfigure demonstrates the process of randomly adding noise to the original region boxes and posing the question to ChatSpot. The right subfigure showcases the performance of ChatSpot after incorporating random noise perturbations.

6 Conclusion

In this work, we first propose *precise referring instruction* tuning for MLLMs that utilizes diverse reference representations for referring special regions. Based on precise referring instruction, we build ChatSpot, a fully end-to-end MLLM that supports diverse region referring prompts, *i.e.*, points, and boxes. Then we construct a large-scale multi-grained vision-language instruction-following dataset, MGVLID. Trained on MGVLID, ChatSpot demonstrates outstanding performance both in interactive chatting and downstream tasks. Results suggest that combining precise referring instructions with MLLMs stimulates the model's ability for special region understanding and reasoning.

Acknowledgements

This work was supported by National Science and Technology Major Project of China (2023ZD0121300).

Contribution Statement

In this work, *Liang Zhao* and *En Yu* are co-first authors, and *Zheng Ge* is the corresponding author.

References

[Alayrac *et al.*, 2022] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.

[Bai *et al.*, 2023] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[Biten *et al.*, 2022] Ali Furkan Biten, Ron Litman, Yusheng Xie, Srikar Appalaraju, and R Manmatha. Latr: Layout-aware transformer for scene-text vqa. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16548–16558, 2022.

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

[Bubeck *et al.*, 2023] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

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

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

[Chiang *et al.*, 2023] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. https://lmsys.org/blog/2023-03-30-vicuna/, 2023.

[Chng *et al.*, 2019] Chee Kheng Chng, Yuliang Liu, Yipeng Sun, Chun Chet Ng, Canjie Luo, Zihan Ni, ChuanMing Fang, Shuaitao Zhang, Junyu Han, Errui Ding, et al. Icdar2019 robust reading challenge on arbitrary-shaped text-rrc-art. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1571–1576. IEEE, 2019.

[Christiano *et al.*, 2017] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

[Dai *et al.*, 2024] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36, 2024.

[Deng *et al.*, 2021] Jiajun Deng, Zhengyuan Yang, Tianlang Chen, Wengang Zhou, and Houqiang Li. Transvg: End-to-end visual grounding with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1769–1779, 2021.

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

[Dong *et al.*, 2023] Runpei Dong, Chunrui Han, Yuang Peng, Zekun Qi, Zheng Ge, Jinrong Yang, Liang Zhao, Jianjian Sun, Hongyu Zhou, Haoran Wei, et al. Dreamllm: Synergistic multimodal comprehension and creation. *arXiv preprint arXiv:2309.11499*, 2023.

[Gan *et al.*, 2020] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. *Advances in Neural Information Processing Systems*, 33:6616–6628, 2020.

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

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

[Krishna *et al.*, 2017] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A

Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017.

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

[Liang *et al.*, 2023] Yaobo Liang, Chenfei Wu, Ting Song, Wenshan Wu, Yan Xia, Yu Liu, Yang Ou, Shuai Lu, Lei Ji, Shaoguang Mao, et al. Taskmatrix. ai: Completing tasks by connecting foundation models with millions of apis. *arXiv preprint arXiv:2303.16434*, 2023.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV*, pages 740–755, 2014.

[Liu *et al.*, 2023a] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023.

[Liu *et al.*, 2023b] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023.

[Long *et al.*, 2022] Shangbang Long, Siyang Qin, Dmitry Panteleev, Alessandro Bissacco, Yasuhisa Fujii, and Michalis Raptis. Towards end-to-end unified scene text detection and layout analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1049–1059, 2022.

[Loshchilov and Hutter, 2019] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[Mani *et al.*, 2020] Arjun Mani, Nobline Yoo, Will Hinthorn, and Olga Russakovsky. Point and ask: Incorporating pointing into visual question answering. *arXiv preprint arXiv:2011.13681*, 2020.

[Mathew *et al.*, 2021] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021.

[Mishra *et al.*, 2019] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *2019 international conference on document analysis and recognition (ICDAR)*, pages 947–952. IEEE, 2019.

[OpenAI, 2023a] OpenAI. Chatgpt. https://openai.com/blog/chatgpt/, 2023.

[OpenAI, 2023b] OpenAI. Gpt-4 technical report, 2023.

[Ouyang *et al.*, 2022] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

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

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

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

[Shao *et al.*, 2019] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019.

[Sharma *et al.*, 2018] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018.

[Shen *et al.*, 2023] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. *arXiv preprint arXiv:2303.17580*, 2023.

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

[Taori *et al.*, 2023] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

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

[Veit *et al.*, 2016] Andreas Veit, Tomas Matera, Lukas Neumann, Jiri Matas, and Serge Belongie. Coco-text: Dataset and benchmark for text detection and recognition in natural images. *arXiv preprint arXiv:1601.07140*, 2016.

[Wang *et al.*, 2022] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jin-

gren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022.

[Wang *et al.*, 2023] Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *arXiv preprint arXiv:2305.11175*, 2023.

[Wei *et al.*, 2022] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

[Wei *et al.*, 2024] Haoran Wei, Lingyu Kong, Jinyue Chen, Liang Zhao, Zheng Ge, En Yu, Jianjian Sun, Chunrui Han, and Xiangyu Zhang. Small language model meets with reinforced vision vocabulary. *arXiv preprint arXiv:2401.12503*, 2024.

[Wu *et al.*, 2023] Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023.

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

[Yang *et al.*, 2023a] Rui Yang, Lin Song, Yanwei Li, Sijie Zhao, Yixiao Ge, Xiu Li, and Ying Shan. Gpt4tools: Teaching large language model to use tools via self-instruction. *arXiv preprint arXiv:2305.18752*, 2023.

[Yang *et al.*, 2023b] Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*, 2023.

[Yu *et al.*, 2023] En Yu, Liang Zhao, Yana Wei, Jinrong Yang, Dongming Wu, Lingyu Kong, Haoran Wei, Tiancai Wang, Zheng Ge, Xiangyu Zhang, et al. Merlin: Empowering multimodal llms with foresight minds. *arXiv preprint arXiv:2312.00589*, 2023.

[Zeng *et al.*, 2022] Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*, 2022.

[Zhang *et al.*, 2022a] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

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

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

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