Omni-Chart-600K: A Comprehensive Dataset of Chart Types for Chart Understanding

Shulei Wang* **Shuai Yang*** **Wang Lin*** **Zirun Guo** **Sihang Cai**
Hai Huang **Ye Wang** **Jingyuan Chen** **Tao Jin†**
Zhejiang University
shuleiwang@zju.edu.cn

Abstract

To address the deficiencies in chart types and the limited scope of chart tasks in existing datasets, we conducted a comprehensive review of current data collection methodologies. By integrating manual annotation with data generation leveraging GPT-4, we developed a dataset that includes 25 diverse chart types and a broad spectrum of tasks, such as data retrieval and mathematical reasoning. Our analysis of existing models revealed that capabilities in information extraction, mathematical reasoning, and understanding of multiple chart types are essential for performing a variety of chart tasks. To overcome the limitations in these areas, we devised a two-stage training strategy and a method for jointly training the vision encoder tailored for multi-type charts. In the first stage, we designed several tasks to enhance the model's general understanding of charts, aligning multimodal large models pre-trained on natural images to chart tasks. To further improve the model's capability to understand various chart tasks and enhance its reasoning abilities, we employed Chain-of-Thought data for training in the second stage. Through two-stage training on our proposed dataset, the pre-trained multimodal large language model achieved state-of-the-art performance across multiple chart understanding tasks, demonstrating the superiority of our data and methods.

1 Introduction

MLLMs (Alayrac et al., 2022; Li et al., 2023; Dai et al., 2024; Zhu et al., 2023; Chen et al., 2023a; Ye et al., 2023; Liu et al., 2024, 2023b; Bai et al., 2023; Achiam et al., 2023; Chen et al., 2023b,c) that leverage the powerful emergent abilities (Wei et al., 2022a) of LLMs are rapidly advancing and have demonstrated remarkable abilities in visual language tasks (Lin et al., 2023a,b). However, MLLMs trained on large-scale natural images often face challenges when it comes to chart-related tasks. Various types of charts serve as highly intuitive visualization mediums that play a crucial role in facilitating the extraction and communication of information from data. For instance, Sankey charts are commonly employed to depict the flow of energy or populations, while parallel coordinates charts are utilized to compare multiple dimensions across several samples. To reduce the gap between natural images and chart images, many chart models have been proposed and MLLMs for chart understanding (Han et al., 2023; Liu et al., 2023a) have attracted much attention.

While existing chart models have made some progress, there are two main limitations. Firstly,existing methods for generating instruction data primarily relied on manual annotation or template generation. Manual annotation approaches (Masry et al., 2022) can capture diverse questions, but the resulting data may contain errors or ambiguities. Approaches that automatically generate instruction data using models, such as those used for the PlotQA (Methani et al., 2020) dataset, may not fully reflect the real-world problems people would ask. The question formats in these datasets tend to be relatively fixed and lack the full diversity seen in human-generated questions. secondly, in real-world scenarios, there is a wider variety of chart types, yet existing chart models perform poorly on tasks involving this greater diversity. However, existing works primarily focus on understanding chart types like bar charts, line charts, and pie charts. Although methods like Chartllama (Han et al., 2023) and MMC (Liu et al., 2023a) have expanded the repertoire of chart types, their performance on a wider range of chart types and tasks remains limited. This shortcoming in comprehending diverse chart data significantly hinders the development of chart understanding in various domains, such as data analysis, medical

*Equal contribution
†Corresponding author

Findings of the Association for Computational Linguistics: NAACL 2025, pages 4051–4069
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

diagnosis, educational technology, business intelligence, and scientific research.

To address the challenges associated with the collection of the aforementioned chart dataset, we propose a highly scalable three-stage multimodal chart data collection method named Omni-Chart. Specifically, we divide our dataset construction process into three stages: chart data generation, chart generation, and instruction data generation. During the chart data generation stage, we prompt GPT-4 (Achiam et al., 2023) to generate the requisite data for various themes and captions based on our predefined chart types. Subsequently, in the chart generation stage, we randomly select this data and employ pyecharts to construct the charts with different settings (*e.g.*, font, legend, and locations). During the instruction data generation stage, we utilize GPT-4 to generate data that aligns with human preferences and exhibits diversity. By prompting GPT-4 to rewrite pre-designed question templates and calling relevant function interfaces to generate answers, this method combines the strengths of human and template generation, ensuring diverse tasks and questions while reducing errors.

Additionally, to address the issue of limited chart types in previous datasets, we introduce Omni-Chart-600K in this paper. This is a large-scale chart understanding instruction tuning dataset that includes a wider variety of chart types, collected using the three-stage method described above. The dataset has a total of 21 different chart type, including boxplot, heatmap, 3D bar, themeriver, multichart, *etc.* and 9 different task types, including math reasoning, code generation, chain-of-thought reasoning, structural understanding, *etc.*

Due to the existing models' weak capabilities in mathematical reasoning and numerical extraction, we employed a two-stage fine-tuning process on a pre-trained multimodal large language model using the dataset we proposed, and additionally enhanced the model's vision encoder. Through a series of experiments, we demonstrated that models trained on our dataset not only perform better on our multi-chart type benchmark but also achieve state-of-the-art results on traditional datasets.

The key contributions of our work are as follows:

- We propose a flexible and scalable method for collecting high-quality chart instruction data. Through our method, we constructed a high-quality, multimodal chart dataset Omni-Chart-600K with multiple chart types and topics.
- We propose a two-stage training strategy for multi-type charts, tailored to enhance multimodal large language models that have been pre-trained. In the first stage, we align the pre-trained models to the chart domain through extensive training across various chart tasks. In the second stage, we leverage Chain-of-Thought data to further boost the models' capabilities in chart understanding and multimodal reasoning.
- With our dataset and method, we train a model Omni-Chart that not only achieves state-of-the-art performance in multi-type chart understanding benchmarks but also state-of-the-art results in existing benchmarks.

2 Related work

2.1 Multimodal learning

Multimodal learning focuses on learning features from different modalities (Lin et al., b). Some previous approaches (Xia et al., 2024; Huang et al., 2024a,b) have utilized unified multimodal representations to align the semantic information across different modalities, while others (Guo et al., 2024a; Jin et al., 2023) have emphasized the comprehensive learning of each modality. Additionally, there are methods (Yan et al., 2025; Guo et al., 2024b) that propose efficient techniques for multimodal learning. Existing approaches (Chen et al., 2023a; Dai et al., 2024) typically leverage the powerful capabilities of large language models, utilizing multimodal data for training.

2.2 Chart-related dataset

Currently, many multimodal chart-related datasets (Huang et al., 2024c; Masry et al., 2022) have been proposed to enhance the chart comprehension capabilities of visual language models. FigureSeer (Siegel et al., 2016) focuses on chart data from scientific papers but is limited to only one type of chart, which is overly simplistic for diverse applications in real world. In contrast, datasets such as FigureQA (Kahou et al., 2017), DVQA (Kafle et al., 2018), LEAF-QA (Chaudhry et al., 2020), Chart-to-text (Kantharaj et al., 2022b), PlotQA (Methani et al., 2020) and LEAFQA++ (Singh and Shekhar, 2020) encompass a broader array of chart types, including bar charts and scatter plots, thereby enhancing data diversity. However, these datasets suffer

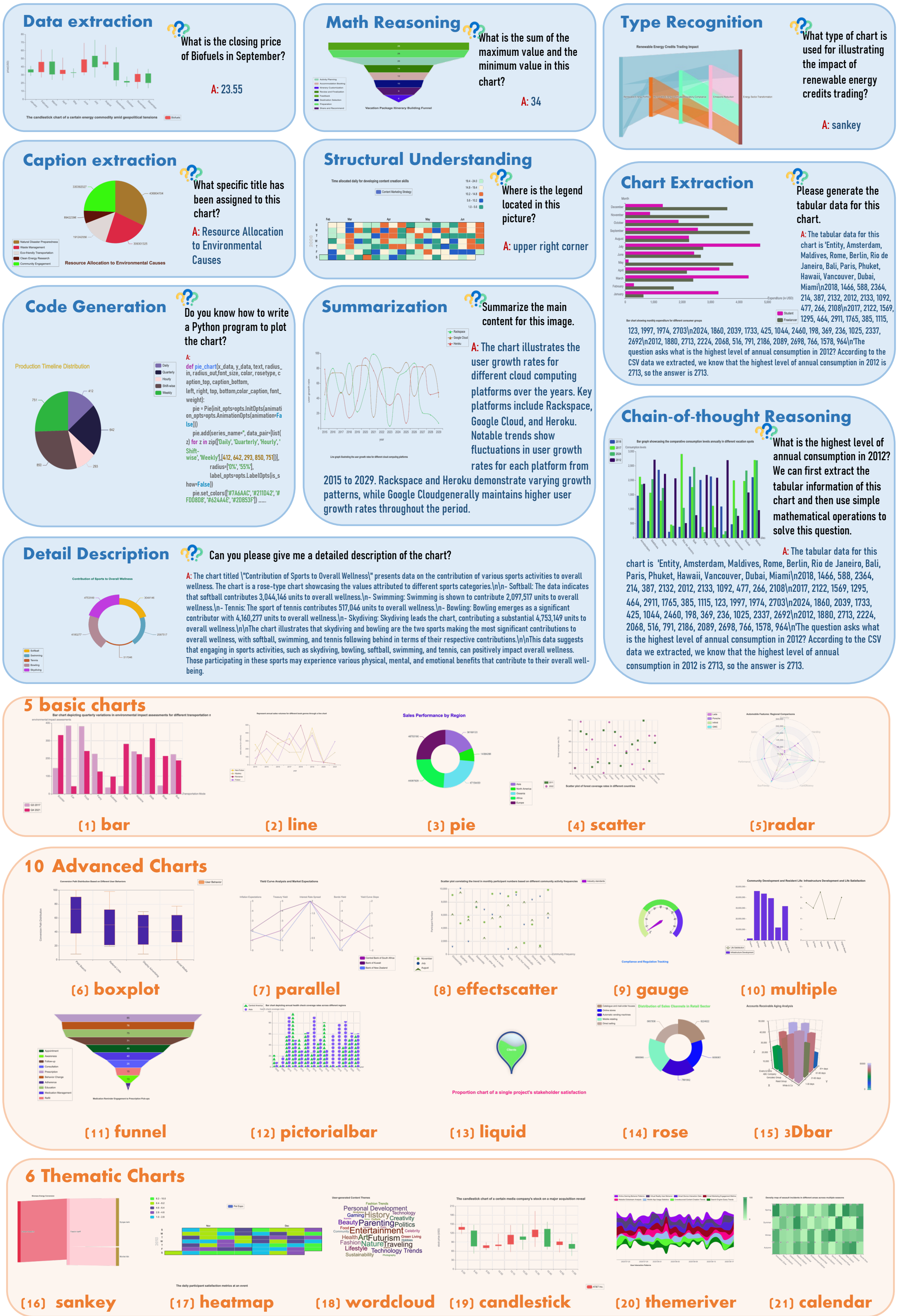


Figure 1: Our instruction dataset Omni-Chart-600K includes 10 types of tasks and 21 types of charts.

datasets	chart type	chart figure	instruction tuning data	task type
chartQA (Masry et al., 2022)	3	21.9K	32.7K	1
plotQA (Methani et al., 2020)	3	224K	28M	1
char-to-text (Kantharaj et al., 2022b)	6	44K	44K	1
Unichart (Masry et al., 2023)	3	627K	7M	3
StructChart (Xia et al., 2023)	3	9K	9K	1
ChartLlama (Han et al., 2023)	10	11K	160K	7
Omni-Chart-600K	**21**	**633K**	**6.8M**	**10**

Table 1: Comparison of our instruction dataset against mainstream datasets related to charts.

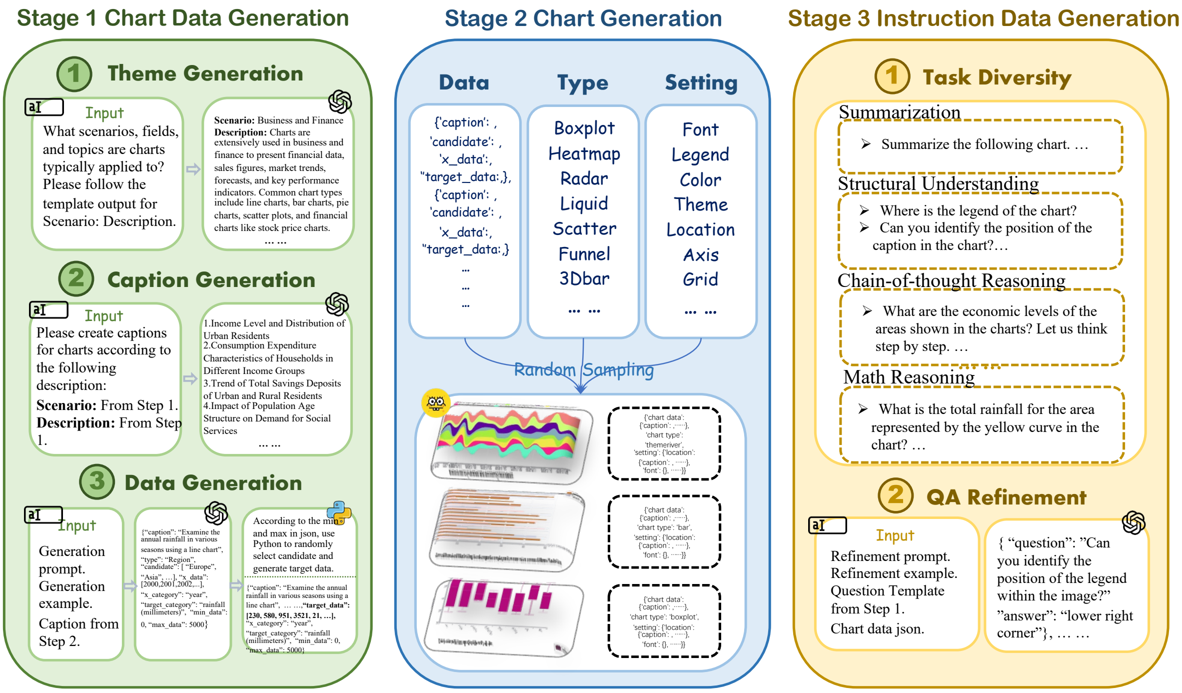


Figure 2: The overall process of our chart dataset construction.

from limitations due to their reliance on template-generated questions and overly simplistic answers derived from a fixed vocabulary. ChartQA (Masry et al., 2022) represents a significant improvement by utilizing manually annotated data and incorporating tasks that demand higher levels of logical and mathematical reasoning, aligning more closely with the complexities of real-world problems. The instructional data proposed by MMC (Liu et al., 2023a) continues to focus on chart types traditionally common in earlier datasets. Chartllama (Han et al., 2023) further diversifies the chart typology to include ten types, introducing less common varieties such as funnel and box charts, thereby broadening the scope for application in more specialized contexts.

2.3 Chart Understanding

Chart data, as an important form of structured data, plays a crucial role in understanding and visualizing real-world data. previous approaches(Kantharaj et al., 2022a; Hsu et al., 2021; Li and Tajbakhsh, 2023; Cheng et al., 2023; Xia et al., 2023; Liu et al., 2022b; Lee et al., 2023) to image understanding can be mainly categorized into two types. The first type involves fine-tuning models that are pretrained on different downstream tasks or datasets. For example, Matcha(Liu et al., 2022b) enhances the pretraining of Pix2Struct(Lee et al., 2023) by designing tasks related to chart derendering and mathematical reasoning, which performs well on tasks such as QA and chart-to-text. Deplot(Liu et al., 2022a) utilizes a modality transformation module trained to convert images into tabular representations, which are then leveraged to exploit the reasoning capabilities of Large Language Models (LLMs) for enhanced comprehension of chart-based tasks.. UniChart(Masry et al., 2023) expands on more pretraining tasks, demonstrating better generalization capabilities. The other approach, such as ChartLlama(Han et al., 2023), fine-tunes the model using constructed instruction data, eliminating the need for task-specific or dataset-specific fine-tuning. However, in the real world, charts are more diverse than the few types primarily focused on by current models (bar charts, line charts, pie charts), and these models often lack the ability to understand other types of charts. Additionally, existing evaluation datasets also lack diversity in chart types. For example, ChartQA(Masry et al., 2022) only includes three types of charts, and the benchmark proposed by MMC(Liu et al., 2023a) lacks a comprehensive evaluation of other chart types, failing to fully assess the chart understanding capabilities of current models

3 Datasets

3.1 Background

Despite the increasing importance of chart understanding for multimodal AI models, existing chart dataset benchmarks such as ChartQA have limitations in their diversity of chart types and the quality of their question-answer annotations. Compared to the wide variety of chart visualizations in the real world, these datasets cover a relatively narrow range of chart categories. Furthermore, the question-answer pairs in these datasets, whether human-annotated or model-generated, often do not fully capture the nuanced and diverse types of questions that humans might naturally ask about data visualizations.

To better drive and evaluate the chart understanding capabilities of large multimodal models, there is a need to develop a more comprehensive dataset that covers a broader spectrum of chart types and contains high-quality question-answer annotations that reflect real-world chart analysis tasks. In this section, we will first describe the process of our chart dataset construction, and then present a thorough analysis of its contents and characteristics.

Our data construction process can be mainly divided into three stages: chart data generation, chart generation, and instruction data generation. The overall process is presented in Figure 2. We will introduce them in the following subsections.

3.2 Chart Data Generation

The goal of the chart data generation stage is to establish a diverse set of chart-related data that can serve as the foundation for building a comprehensive chart dataset for large multimodal model training. Unlike approaches that leverage the capabilities of image generation models (Lin et al., 2024, a), we divide the chart data generation stage into three distinct steps, leveraging the capabilities of GPT-4 to generate a wide range of chart themes, captions, and data. More details about the prompts can be found in the appendix A

Step 1: Theme Generation. The first step involves leveraging GPT-4 to generate a diverse set of scenarios, fields, and topics where charts are typically applied. By prompting GPT-4 with the question, the model will produce a set of scenarios, each accompanied by a descriptive paragraph. This step aims to establish a broad range of chart themes that can be used as the foundation for the subsequent caption and data generation.

Step 2: Caption Generation. Building upon the themes generated in the previous step, step 2 utilizes GPT-4 to create captions for charts corresponding to each of the identified themes. For each theme and its accompanying description, the model is prompted to "Please create captions for charts according to the following description: Scenario: [Scenario from Step 1], Description: [Description from Step 1]." Notably, for each theme, N captions will be generated. This step ensures that the captions are tailored to the specific contexts, enhancing the relevance and diversity of the dataset.

Step 3: Data Generation. For each caption generated in the previous step, we will request GPT-4 to generate a JSON object containing several keys, such as the caption itself, the type of chart, a list of candidate data points, and the minimum and maximum values for the data points. Since GPT-4 often generates duplicate and redundant data, in order to further enhance the diversity of the data and make it more realistic, we create a Python program to randomly select a subset of the candidate data points and generate target data within the specified ranges, ensuring a diverse set of chart types, data distributions, and scales.

3.3 Chart Generation

The goal of this stage is to generate a diverse set of chart figures based on the chart data generated in stage 1. This stage utilizes Pyecharts to plot

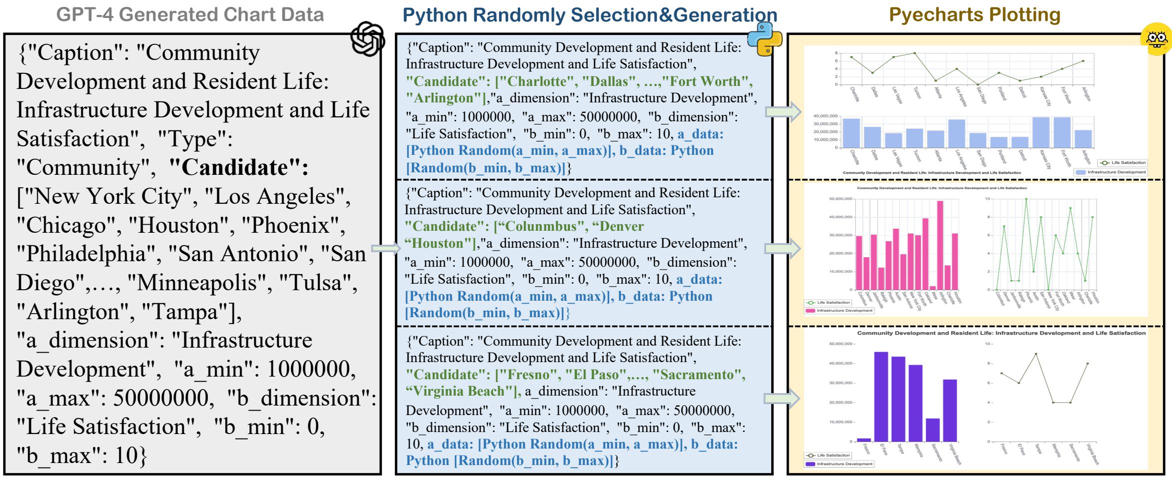


Figure 3: The pipeline of chart generation. We first utilize GPT-4 to generate candidates and the value range of target data. Then we use Python to randomly select several candidates (denoted in Green) and generate target data (denoted in Blue). Finally, we use Pyecharts to plot the chart with random plot settings (*e.g.* color, legend). We use multichart generation as an example.

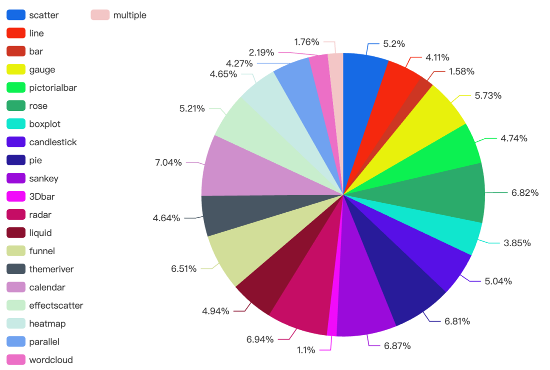


Figure 4: Distribution of our datasets.

chart figures, ensuring a realistic and diverse set of visualizations. Pyecharts supports a wide range of chart types, enabling us to produce charts of various categories based on the diverse data obtained from stage 1.

The chart generation process begins by inputting the chart data generated in stage 1, along with a randomly selected chart type from a total of 21 types (*e.g.*, bar, heatmap, liquid). Additionally, a chart setting is randomly selected, which includes various customizable elements such as the location of the legend, caption, font, color, and more. This random selection ensures that the generated charts are diverse and realistic, mimicking the variability in real-world visualizations. In Figure 3, we present the process of chart generation of an example.

3.4 Instruction Data Generation

The goal of this stage is to generate high-quality instruction data that addresses the limitations of previous methods. Concretely, previous methods for generating instruction data primarily relied on manual annotation and template generation, both of which have their own challenges. Firstly, manual annotation, despite showcasing the diversity of questions, often contains errors. For example, in the ChartQA dataset, instructions include relative terms like "last year", which not only lack complete information for model training but also hinder the assessment of the model's chart comprehension abilities. Similarly, methods like those used in PlotQA, which construct instruction data using models, exhibit a gap from the real-world problems people would actually ask, with the question format being relatively fixed and lacking variety. Based on the above observation, we divide our instruction data generation into two steps to ensure task and question diversity, respectively. In the first step, manual design of tasks for charts is performed to ensure task diversity. We design a total of 10 task types, such as summarization questions, detail understanding questions, chain-of-thought reasoning, and math reasoning as shown in Figure 8. In the second step, the task template designed in step 1 is input into GPT-4, along with an example and chart data in JSON format from stage 2. GPT-4 is then used for question refinement, making the

Dataset	Understandability	Relevance	Clarity	Reasonableness	Correctness	Average
ChartQA (Masry et al., 2022)	**2.52**	2.29	**2.67**	2.63	2.87	2.596
Omni-Chart-600K	2.49	**2.34**	2.61	**2.65**	**2.90**	**2.598**

Table 2: User study of different datasets based on multiple criteria.

task question more fitting to the specific chart. This step refines the task questions to ensure they are well-suited to the chart data, resulting in a more diverse and realistic set of instruction data.

3.5 Data statistic

Quantitative Analysis: We present the statistics of our generated dataset in Table 1 and Figure 4. Our dataset comprises 21 different types of chart data, making it, to the best of our knowledge, the dataset with the most chart types included. In addition to common chart forms such as bar charts, line charts, and pie charts, our dataset also encompasses 3D bar charts, parallel charts, theme river charts, Sankey charts, and more, as shown in Figure 1. Additionally, our dataset distinguishes itself from others by maintaining a balanced distribution across all chart types, in contrast to traditional datasets where basic chart types predominate. Regarding task variety, our dataset is exceptionally comprehensive. Furthermore, drawing inspiration from (Wei et al., 2022b), we have meticulously designed tasks for Chain of Thoughts reasoning. These tasks are specifically tailored to bolster the chart reasoning capabilities of models, an essential attribute for practical applications in the real world.

Quantitative Analysis: we conducted a comprehensive user study on data quality with 10 participants. For each chart type, we randomly selected 10 different chart QA pairs, making a total of 210 chart QA pairs. Each participant was asked to answer five multiple-choice questions for each QA pair, evaluating the data across five aspects. For example, Relevance measures the correlation between the question and the chart data, while Reasonableness evaluates whether the question aligns with real-world scenarios. As shown in Table 2, the user study indicates that our data quality surpasses that of ChartQA.

4 Experiments

4.1 Architecture baseline

Based on the LLaVA(Liu et al., 2024) framework, we develop our Omni-Chart model for chart understanding. LLaVA, which has been pretrained on a diverse array of visual language tasks, brings a wealth of prior knowledge that is crucial for large language models (LLMs). To effectively capture both the overarching and detailed features of charts, the input chart is initially resized into a single patch. This is complemented by additional patches that are created from a high-resolution version of the image. This dual approach ensures a comprehensive analysis of the visual data. In the input sequence, the image tokens are strategically placed at the start, immediately followed by the language query. To validate the effectiveness of the dataset we constructed, Omni-Chart-600K, we established a baseline based on LLaVA1.6.

4.2 Training Details

To enhance the capabilities of our multimodal large language model, we have developed a two-phase training pipeline. In the first phase, we train the pretrained MLLM model using simple tasks such as data extraction and structural understanding to align it with chart-related tasks. In the second phase, we employ more complex tasks, such as mathematical reasoning and chain of thoughts reasoning, to unleash the potential of the MLLM. Furthermore, through our analysis comparing chart images with natural images, we observed that chart images usually contain finer-grained numerical data and legends. Consequently, we employed a method of jointly fine-tuning the LLaVA vision encoder to develop the Omni-Chart model. Throughout this training, we employ LoRA (Hu et al., 2021) with a rank of 32. In the second stage, we integrate the LoRA weights from the previous stage and introduce a new LoRA adapter to optimize performance.

4.3 Evaluation benchmark and metric

Tasks: To comprehensively assess the performance of our model, we employ three established downstream tasks. ChartQA (Masry et al., 2022), where performance is evaluated by relaxed accuracy on human and augmentation splits. The human split presents a more challenging dataset as it includes

model	chart QA			Chart-to-Table	Chart-to-Text	
	aug	human	overall	Chart QA	Pew	Statista
Pix2struct (Lee et al., 2023)	81.6	30.5	56.0	-	10.3	38.0
Matcha (Liu et al., 2022b)	90.2	38.2	64.2	-	12.20	39.40
DePlot (Liu et al., 2022a)	-	-	-	79.3	-	-
Chart-T5 (Zhou et al., 2023a)	74.4	31.8	53.1	-	9.10	37.5
Unichart (Masry et al., 2023)	87.8	43.9	65.85	91.1	12.5	38.1
Chartllama (Han et al., 2023)	90.4	48.9	69.65	90.0	14.2	40.7
Omni-chart (Ours)	**86.6**	**60.9**	**73.8**	**92.2**	**15.8**	**53.2**

Table 3: Results on several benchmarks

questions requiring mathematical reasoning. Chart-to-Table (Liu et al., 2022a), which requires the model to extract CSV data from charts. Performance is measured using RMS_{F1} from DePlot. Chart-to-Text (Kantharaj et al., 2022b), focusing on converting chart data directly into descriptive text. Additionally, to verify the existing models' understanding of multiple types of charts, we constructed a benchmark Omni-ChartQA for understanding multiple chart types using the method we proposed. This benchmark includes QA tasks for 21 types of charts and employs relaxed accuracy as the evaluation metric, comprehensively assessing the models' ability to understand a wider variety of chart types.

Baselines: We benchmark the performance of Omni-Chart model against seven baseline models: LLaVA (Liu et al., 2024), Chart-T5 (Zhou et al., 2023b), Chartllama (Han et al., 2023), UniChart (Masry et al., 2023), cogvlm-chat-hf (Wang et al., 2023), deepseek-vl (Lu et al., 2024), internVL2-8B (Chen et al., 2024), Qwen2-VL-7B-Instruct (Wang et al., 2024).

4.4 Results

As demonstrated in Table 3, the multimodal large language models trained using our dataset have achieved state-of-the-art performance across several benchmarks. Our models particularly excel on the ChartQA benchmark, which involves complex mathematical reasoning, due to the rich diversity of reasoning information contained within our Chain-of-Thought data. Additionally, as our dataset encompasses a variety of tasks, including data extraction, our methodology secures state-of-the-art results on benchmarks such as Chart-to-Text and Chart-to-Table. Furthermore, as shown in Table 4, general multimodal large language models have limited capability in understanding multiple types of chart data. In contrast, our method significantly enhances the understanding of various chart types, addressing the deficiencies of existing methods in handling a broader spectrum of chart types. Our empirical evidence demonstrates that training large language models on a diverse range of chart types and tasks can substantially improve their ability to effectively understand and interpret various chart formats.

4.5 User Study

Similarly, to evaluate the model's output, we randomly selected 100 chart-question model output pairs. Each participant was required to answer five multiple-choice questions for each QA pair, evaluating five aspects of the model. Each option represents a different score. Through a comprehensive user study, we demonstrated that our model's output performs well across various aspects, as shown in Table 5.

4.6 Ablation Study

We evaluate our design using five distinct ablation settings described in Talbe 6. 'Zero Shot' indicates the use of a pretrained LLaVA model without additional training. 'Direct Fine-Tuned' involves fine-tuning the LLaVA Language Model with LoRA and our QA dataset without Chain of Thought(CoT). 'LLM Only + CoT' consists of fine-tuning the Large Language Model (LLM) exclusively with CoT data. The settings 'LLM + Vision Encoder + CoT' and 'LLM + Vision Encoder* + CoT' assess the effects of varying LoRA ranks, with both involving simultaneous tuning of the LLM and Vision Encoder. Experimental results demonstrate that our chart-specific Chain of Thought data significantly enhances overall performance, and further tuning of the vision encoder

Chart Type	cogvlm	deepseek	internVL2	Qwen2-VL	Chartllama	LLaVA	Ours
Overall	23.68	19.82	27.04	35.60	8.87	22.89	**44.88**
radar	14.75	11.26	21.72	**28.15**	5.93	15.92	21.87
parallel	30.23	21.40	35.35	**41.86**	5.95	30.34	41.07
rose	17.24	14.48	33.10	47.13	7.80	28.35	**53.48**
pie	14.38	12.36	30.34	40.90	8.47	23.86	**46.79**
wordcloud	34.69	**36.73**	18.37	**36.73**	34.23	20.12	18.79
themeriver	14.62	5.66	17.45	20.28	5.53	19.27	**50.69**
candlestick	2.69	0.45	3.59	6.28	4.90	**16.36**	9.70
sankey	25.57	19.09	27.18	32.04	10.16	24.13	**55.25**
effectscatter	26.33	23.13	32.03	42.70	8.28	18.83	**38.85**
bar	23.17	21.95	30.49	**40.24**	9.20	17.77	34.17
pictorialbar	29.35	22.89	25.37	**36.32**	14.71	20.91	34.04
boxplot	9.39	17.84	14.08	13.62	6.48	26.40	**47.20**
heatmap	39.22	33.19	49.57	52.16	17.40	26.76	**55.38**
gauge	38.62	27.24	31.71	47.56	7.77	28.10	**52.08**
scatter	26.94	31.84	34.29	**46.53**	10.81	19.07	34.90
funnel	35.03	23.05	40.72	62.57	5.57	27.11	**82.94**
multiple	22.58	22.58	17.74	**29.84**	9.80	14.68	27.32
line	26.17	18.22	31.31	**34.58**	8.31	17.89	28.78
calendar	21.12	20.50	28.26	22.05	8.55	29.32	**76.25**
Liquid Fill Chart	30.06	23.31	26.99	**41.72**	6.34	16.65	20.48
3Dbar	15.15	9.09	18.18	**24.24**	8.69	22.79	19.67

Table 4: Results by different chart types on Omni-ChartQA.

Dataset	Accuracy	Relevance	Readability	Reasonableness	Completeness	Average
Chartllama (Han et al., 2023)	2.31	2.42	**2.63**	2.32	2.57	2.450
Omni-Chart-600K	**2.55**	**2.45**	2.58	**2.34**	**2.60**	**2.504**

Table 5: User study on the outputs of different models.

method	ChartQA overall
Zero shot	47.3
direct fine-tuned	51.4
LLM Only + CoT	71.0
LLM+Vison Encoder + CoT	71.8
LLM+Vison Encoder*+CoT	**73.8**

Table 6: Ablations study: '*' indicates the use of a LoRA rank of 32, while the other configurations employ a LoRA rank of 16.

also improves scores, corroborating the findings in (Laurençon et al., 2024).

5 Conclusion

In this paper, we introduce a versatile and scalable three-stage instruction collection method, significantly reducing labeling errors and increasing data diversity. Utilizing this methodology, we developed Omni-Chart-600K, which, to the best of our knowledge, is the most comprehensive chart instruction dataset in terms of chart variety, comprising 21 distinct chart categories and 10 task types, with a total of 633K chart images and 6.8M instruction data. Additionally, we present a two-stage training strategy that leverages fine-tuning on a pretrained multimodal large language model, achieving SOTA outcomes not only in traditional chart understanding tasks but also in more challenging tasks that demand comprehensive comprehension of multiple diverse chart types. We firmly posit that our dataset and methodological approach significantly enhance MLLM's capability to accurately interpret and analyze the vast spectrum of complex charts encountered in real-world scenarios.

6 Limitations

In our exploration of automated chart data process, we introduced a dataset comprising 21 distinct types of charts. While this diversity represents a significant advancement in the field, some limitations have been identified that could impact the effectiveness of models trained on this dataset. Concretely, although the dataset encompasses a broad array of chart types, it does not cover the entire spectrum of charts found in real-world applications. This gap in the dataset might hinder the model's ability to generalize well across truly diverse chart types, potentially limiting its applicability in practical scenarios where uncommon or complex charts are prevalent. Addressing these limitations represents significant future work.

7 Acknowledgments

We sincerely thank the anonymous reviewers for their constructive feedback. This research was supported by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang under Grant No. 2025C02022.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Ritwick Chaudhry, Sumit Shekhar, Utkarsh Gupta, Pranav Maneriker, Prann Bansal, and Ajay Joshi. 2020. Leaf-qa: Locate, encode & attend for figure question answering. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3512–3521.

Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. 2023a. Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*.

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*.

Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. 2023c. Sharegpt4v: Improving large multimodal models with better captions. *arXiv preprint arXiv:2311.12793*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Zhi-Qi Cheng, Qi Dai, and Alexander G Hauptmann. 2023. Chartreader: A unified framework for chart derendering and comprehension without heuristic rules. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22202–22213.

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

Zirun Guo, Tao Jin, Wenlong Xu, Wang Lin, and Yangyang Wu. 2024a. Bridging the gap for test-time multimodal sentiment analysis. *arXiv preprint arXiv:2412.07121*.

Zirun Guo, Tao Jin, and Zhou Zhao. 2024b. Multimodal prompt learning with missing modalities for sentiment analysis and emotion recognition. *arXiv preprint arXiv:2407.05374*.

Yucheng Han, Chi Zhang, Xin Chen, Xu Yang, Zhibin Wang, Gang Yu, Bin Fu, and Hanwang Zhang. 2023. Chartllama: A multimodal llm for chart understanding and generation. *arXiv preprint arXiv:2311.16483*.

Ting-Yao Hsu, C Lee Giles, and Ting-Hao'Kenneth' Huang. 2021. Scicap: Generating captions for scientific figures. *arXiv preprint arXiv:2110.11624*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Hai Huang, Shulei Wang, and Yan Xia. 2024a. Semantic residual for multimodal unified discrete representation. *arXiv preprint arXiv:2412.19128*.

Hai Huang, Yan Xia, Shengpeng Ji, Shulei Wang, Hanting Wang, Jieming Zhu, Zhenhua Dong, and Zhou Zhao. 2024b. Unlocking the potential of multimodal unified discrete representation through training-free codebook optimization and hierarchical alignment. *arXiv preprint arXiv:2403.05168*.

Zihan Huang, Tao Wu, Wang Lin, Shengyu Zhang, Jingyuan Chen, and Fei Wu. 2024c. Autogeo: Automating geometric image dataset creation for enhanced geometry understanding. *arXiv preprint arXiv:2409.09039*.

Tao Jin, Xize Cheng, Linjun Li, Wang Lin, Ye Wang, and Zhou Zhao. 2023. Rethinking missing modality learning from a decoding perspective. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 4431–4439.

Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. 2018. Dvqa: Understanding data visualizations via question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5648–5656.

Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Ákos Kádár, Adam Trischler, and Yoshua Bengio. 2017. Figureqa: An annotated figure dataset for visual reasoning. *arXiv preprint arXiv:1710.07300*.

Shankar Kantharaj, Xuan Long Do, Rixie Tiffany Ko Leong, Jia Qing Tan, Enamul Hoque, and Shafiq Joty. 2022a. Opencqa: Open-ended question answering with charts. *arXiv preprint arXiv:2210.06628*.

Shankar Kantharaj, Rixie Tiffany Ko Leong, Xiang Lin, Ahmed Masry, Megh Thakkar, Enamul Hoque, and Shafiq Joty. 2022b. Chart-to-text: A large-scale benchmark for chart summarization. *arXiv preprint arXiv:2203.06486*.

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. 2024. What matters when building vision-language models? *arXiv preprint arXiv:2405.02246*.

Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. 2023. Pix2struct: Screenshot parsing as pretraining for visual language understanding. In *International Conference on Machine Learning*, pages 18893–18912. PMLR.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pretraining with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Shengzhi Li and Nima Tajbakhsh. 2023. Scigraphqa: A large-scale synthetic multi-turn question-answering dataset for scientific graphs. *arXiv preprint arXiv:2308.03349*.

Wang Lin, Jingyuan Chen, Jiaxin Shi, Zirun Guo, Yichen Zhu, Zehan Wang, Tao Jin, Zhou Zhao, Fei Wu, YAN Shuicheng, et al. a. Action imitation in common action space for customized action image synthesis. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Wang Lin, Jingyuan Chen, Jiaxin Shi, Yichen Zhu, Chen Liang, Junzhong Miao, Tao Jin, Zhou Zhao, Fei Wu, Shuicheng Yan, et al. 2024. Non-confusing generation of customized concepts in diffusion models. *arXiv preprint arXiv:2405.06914*.

Wang Lin, Yueying Feng, WenKang Han, Tao Jin, Zhou Zhao, Fei Wu, Chang Yao, and Jingyuan Chen. b. E3: Exploring embodied emotion through a large-scale egocentric video dataset. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Wang Lin, Tao Jin, Wenwen Pan, Linjun Li, Xize Cheng, Ye Wang, and Zhou Zhao. 2023a. Tavt: Towards transferable audio-visual text generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14983–14999.

Wang Lin, Tao Jin, Ye Wang, Wenwen Pan, Linjun Li, Xize Cheng, and Zhou Zhao. 2023b. Exploring group video captioning with efficient relational approximation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15281–15290.

Fangyu Liu, Julian Martin Eisenschlos, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Wenhu Chen, Nigel Collier, and Yasemin Altun. 2022a. Deplot: One-shot visual language reasoning by plot-to-table translation. *arXiv preprint arXiv:2212.10505*.

Fangyu Liu, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Yasemin Altun, Nigel Collier, and Julian Martin Eisenschlos. 2022b. Matcha: Enhancing visual language pretraining with math reasoning and chart derendering. *arXiv preprint arXiv:2212.09662*.

Fuxiao Liu, Xiaoyang Wang, Wenlin Yao, Jianshu Chen, Kaiqiang Song, Sangwoo Cho, Yaser Yacoob, and Dong Yu. 2023a. Mmc: Advancing multimodal chart understanding with large-scale instruction tuning. *arXiv preprint arXiv:2311.10774*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023b. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, et al. 2024. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*.

Ahmed Masry, Parsa Kavehzadeh, Xuan Long Do, Enamul Hoque, and Shafiq Joty. 2023. Unichart: A universal vision-language pretrained model for chart comprehension and reasoning. *arXiv preprint arXiv:2305.14761*.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv preprint arXiv:2203.10244*.

Nitesh Methani, Pritha Ganguly, Mitesh M Khapra, and Pratyush Kumar. 2020. Plotqa: Reasoning over scientific plots. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1527–1536.

Noah Siegel, Zachary Horvitz, Roie Levin, Santosh Divvala, and Ali Farhadi. 2016. Figureseer: Parsing result-figures in research papers. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pages 664–680. Springer.

Hrituraj Singh and Sumit Shekhar. 2020. Stl-cqa: Structure-based transformers with localization and encoding for chart question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3275–3284.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, et al. 2023. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022b. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Renqiu Xia, Bo Zhang, Haoyang Peng, Ning Liao, Peng Ye, Botian Shi, Junchi Yan, and Yu Qiao. 2023. Structchart: Perception, structuring, reasoning for visual chart understanding. *arXiv preprint arXiv:2309.11268*.

Yan Xia, Hai Huang, Jieming Zhu, and Zhou Zhao. 2024. Achieving cross modal generalization with multimodal unified representation. *Advances in Neural Information Processing Systems*, 36.

Weicai Yan, Wang Lin, Zirun Guo, Ye Wang, Fangming Feng, Xiaoda Yang, Zehan Wang, and Tao Jin. 2025. Diff-prompt: Diffusion-driven prompt generator with mask supervision. In *The Thirteenth International Conference on Learning Representations*.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. 2023. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Mingyang Zhou, Yi R Fung, Long Chen, Christopher Thomas, Heng Ji, and Shih-Fu Chang. 2023a. Enhanced chart understanding in vision and language task via cross-modal pre-training on plot table pairs. *arXiv preprint arXiv:2305.18641*.

Mingyang Zhou, Yi R Fung, Long Chen, Christopher Thomas, Heng Ji, and Shih-Fu Chang. 2023b. Enhanced chart understanding in vision and language task via cross-modal pre-training on plot table pairs. *arXiv preprint arXiv:2305.18641*.

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 Prompt design

Our approach leverages GPT-4 to generate high-quality chart data. To fully unleash the generative capabilities of GPT-4 and minimize errors, we have designed sophisticated prompts. Specifically, to produce chart data of varied themes and types, we initially prompt GPT-4 to generate application themes for specific types of charts. Subsequently,as shown in the Figure 5, GPT-4 is prompted to create 100 titles for each specific chart type, applying a pre-generated theme every 50 titles to enhance the diversity of the chart data. To generate the data constituting the charts, as shown in the Figure 6,we employ a one-shot approach where GPT-4 is given an example to enhance its instruction-following capabilities, thus ensuring the generation of accurate and diverse chart data. Similarly, in the question rewriting phase,as shown in the Figure 7, we utilize a one-shot method to enrich the format of the questions, making them more representative of real-world scenarios.

B template

Our method combines the strengths of human annotation and template generation to produce high-quality chart instruction data. We posit that the range of questions about charts in the real world is finite. Consequently, as illustrated in Figure 8, we have designed comprehensive template questions that address a variety of tasks such as identifying maximum and minimum values, the positions of legends, and simple mathematical reasoning. These template questions encompass all potential tasks and have been diversified through rewriting by GPT-4. The resultant diverse instruction data substantially supports multimodal large language models in better understanding chart data, enhancing their performance across relevant tasks.

C Qualitative Results

C.1 Results of chartQA

Reasoning capability is a crucial aspect of multimodal large language models. To better harness this ability, we have designed tasks based on chain-of-thought reasoning, enabling the model to progressively reason through complex questions to arrive at the final answer. As shown in Figure 9, ChartQA is a challenging dataset that involves diverse reasoning tasks. Compared to ChartLlama, our model achieves more accurate results through COT-based reasoning, demonstrating its enhanced capability to handle intricate problem-solving scenarios.

C.2 Results of Chart-to-Table

The task of Chart-to-Table involves parsing charts into CSV data. We have also incorporated a Chart-to-Table task during training and have parsed various types of charts. This approach has significantly enhanced the model's parsing capabilities, as demonstrated in Figure 10.

C.3 Results of Chart-to-Text

Through comprehensive multitask training, our model also performs better on the Chart-to-Text task.

D Baseline Model Details

D.1 Pix2Struct

Pix2Struct is a pretrained image-to-text model designed to understand visually-situated language, which encompasses a wide range of visual sources such as textbooks with diagrams, web pages with images and tables, and mobile apps with buttons and forms. The model addresses the limitations of previous domain-specific approaches that lacked flexibility and shared resources by introducing a novel pretraining strategy.

Pix2Struct's pretraining involves parsing masked screenshots of web pages into simplified HTML, leveraging the rich visual elements found on the web to generate a diverse pretraining dataset. This method not only utilizes traditional pretraining signals such as OCR, language modeling, and image captioning but also enhances them by translating complex visual information into structured HTML text.

D.2 Matcha

Matcha builds upon the foundation of Pix2Struct. It introduces several pretraining tasks specifically tailored to improve the model's capabilities in plot deconstruction and numerical reasoning, which are critical for effective visual language modeling.

D.3 DePlot

Recognizing the limitations of previous state-of-the-art models, which require extensive training on large datasets and still show limited reasoning capabilities, especially with complex human-written queries, DePlot presents a significant advancement by offering a few-shot solution for visual language reasoning.

DePlot's methodology decomposes the challenge into two primary steps:

1. Plot-to-Text Translation: This involves translating the visual information from plots or charts into a linearized table format.

2. Reasoning Over Translated Text: Utilizing the table format to enable reasoning, which leverages the capabilities of large language models (LLMs).

A key innovation in DePlot is the modality conversion module that performs the plot-to-table translation. This output is then used to prompt a pretrained large language model, harnessing the few-shot reasoning capabilities of these models.

D.4 Chart-T5

Chart-T5 operates through a cross-modal pre-training approach, utilizing plot-table pairs to train the model. The pre-training incorporates two novel objectives:

1. Masked Header Prediction (MHP): This objective trains Chart-T5 to recognize and predict the headers of tables derived from chart images, which is crucial for understanding the structure and categorization within charts.

2. Masked Value Prediction (MVP): This trains the model to predict the numeric or textual data within the table cells, enhancing its ability to accurately interpret the detailed information presented in charts.

D.5 Unichart

This model addresses the limitations of existing methods in chart-based data analysis tasks, such as chart question answering and chart summarization, which often rely solely on pretraining in language or vision-language tasks without adequately modeling the explicit structures of charts.

To tackle these challenges, the first step involved building a large corpus of charts that vary in topics and visual styles. UniChart then utilizes this corpus to effectively encode textual, data, and visual elements of charts, employing a chart-grounded text decoder for generating text.

Unichart's pretraining involves a mix of chart-specific tasks designed to improve both low-level and high-level understanding of charts:

Low-level tasks focus on extracting visual elements and data directly from the charts. High-level tasks aim to enhance the model's overall chart understanding and reasoning capabilities.

D.6 LLaVA

LLaVA is one of the most advanced open-source multimodal large language models currently available. LLaVA employs a two-stage training process, where the first stage aligns the visual and language modalities, and the second stage fine-tunes the model for visual instructions. The model leverages machine-generated data where GPT-4 provides language instructions aligned with images, allowing LLaVA to learn from this synthesized multimodal context.

D.7 chartllama

chartllama introduce a novel high-quality instruction-tuning dataset was created using GPT-4. This involved a multi-step data generation process:

1. Tabular Data Generation: This step focused on creating the underlying data that would be visualized in the charts.

2. Chart Figure Creation: Using the tabular data, various chart figures were generated, encompassing a diverse range of chart types.

3. Instruction Tuning Data Design: This final step involved crafting instruction sets that would guide the model in interpreting the charts correctly.

Prompt design

Your current task is to provide titles for pie charts in different application scenarios. Please provide me 100 different titles,requiring 50 of them to be the titles of the pie chart in the Marketing scenario, and the other 50 to be the titles of the pie chart in the Human Resources scenario.
The title is required to summarize related types, such as countries, movies, electronic products, etc.
Here are an example:
"Market share chart of different electronic products"

Figure 5: Prompt for generating titles. The blue ones are pre-generated possible themes for that type of chart, each time generating 100 chart titles of different themes.

Prompt design

Your task is to output a JSON file containing six keys: "caption," "type," "candidate," "characteristic," "min_data," and "max_data." For the given radar chart titles, first output the title, then indicate the object type described by this radar chart. After that, output 100 candidate options for this question. The candidates should be real and unique, and should not include identifiers such as A, B, C, 1, 2, 3, etc. Additionally, provide the nature of the comparison of the candidates in the radar chart. Also, specify the range of values for each characteristic of the radar chart corresponding to the title.
Make sure that the number of items in the "characteristic" key's list is greater than 6.
Here's a simple example:
Input: "Comparison of Different Features of Various Mobile Phone Brands Radar Chart"
Output:
{ "caption": "Comparison of Different Features of Various Mobile Phone Brands Radar Chart", "type": "Mobile Phone Brands", "candidate": ["Apple", "Samsung", "Huawei", "Xiaomi", "Oppo", "Vivo", "OnePlus", "Google (Pixel)", "Sony", "LG", "Nokia", "Motorola", "HTC", "Asus", "Lenovo", "ZTE", "BlackBerry", "Alcatel", "Meizu", "Realme", "Infinix", "Tecno", "Coolpad", "LeEco (Letv)", "Sharp", "Panasonic", "Micromax", "Karbonn", "Lava", "Gionee", "Honor", "Ulefone", "Elephone", "Vernee", "Doogee", "BLU", "Razer", "Wiko", "Essential", "Fairphone", "YotaPhone", "Jolla", "Cat", "HP", "Dell", "Fujitsu", "Casio", "Acer", "Kyocera", "Hisense"], "characteristic": ["Portability", "Price", "Value for Money", "Speed", "Appearance", "User-Friendliness", "Performance", "Practicality", "Durability"], "min_data": 200000, "max_data": 100000000 }
Now, input:"
Radar chart comparing different economic indicators in different countries"
Output:

Figure 6: Prompts for generating chart data. This phase produces all the data necessary for plotting charts, including candidates, characteristics, maximum values, minimum values, and more. The one-shot prompt method ensures the generated data is both accurate and diverse.

Prompt design

Your task is to output a list of JSON and do not output other information, where each dictionary in the list contains four keys named "question" ,"answer","image_index" and "type". You need to combine your data in template form to generate the corresponding question. Then, rewrite that question to make it more varied in form while keeping the original meaning unchanged, and output the question along with the answer,image_index and type. Below are the templates for the questions:
"template": "Does the graph contain grids?"
Here is a simple example:
"input": [{
"caption": "Line Chart of Changes in Song Rankings on Different Music Charts",
"answer": "Yes",
"image_index": 6,
"type": "line"
}]
"output": [{
"question": "Does the line chart depicting changes in song rankings on different music charts include gridlines?",
"answer": "Yes",
"image_index": 6,
"type": "line"
}]
Now, for the current input:
"input": ["{"caption": "Bar chart of monthly sales figures for different product categories", "anwser": "Yes", "image_index": 0, "type": "bar"}", "{"caption": "Bar chart illustrating monthly expenditure for different customer segments", "anwser": "Yes", "image_index": 1, "type": "bar"}", "{"caption": "Bar graph showing monthly returns on different financial instruments", "anwser": "Yes", "image_index": 2, "type": "bar"}", "{"caption": "Bar chart illustrating annual profit margins for different competitors", "anwser": "Yes", "image_index": 3, "type": "bar"}", "{"caption": "Bar chart depicting monthly performance metrics for different sales teams", "anwser": "No", "image_index": 4, "type": "bar"}", "{"caption": "Bar graph representing quarterly market shares of different product brands", "anwser": "No", "image_index": 5, "type": "bar"}”]
"output":

Figure 7: Prompts for rewriting template questions. To generate more diverse instruction data, we prompt GPT-4 to rewrite predetermined template questions. The rewritten questions become more varied, including alternative formulations of generic questions and questions that incorporate chart titles, aligning more closely with real-world scenarios.

template

What is the difference between the maximum value and the minimum value in this chart?
What is the ratio of the maximum value to the minimum value in this picture?
What is the sum of the maximum value and the minimum value in this chart?
What is the maximum value in this chart?
What is the minimum value in this chart?
How many legend labels are there?
How many figure-types are there?
Which item has the largest value at a certain point?
What is the worst performing item on a certain indicator?
What is the chart type of this picture?
What is the closing price of an item at a certain point?
Does the graph contain grids?
Does the graph contain legend?
Does the graph contain a title?
Which data does this picture describe?
What is the pointer pointing to in this picture?
What is the title of the graph?
What is the value of a certain item at a certain point?
what is the proportion of a certain item?
What is the value of a certain item?
How many days of data does this calendar graph represent?
What is the end time of this calendar chart?
What is the best indicator of performance for a given item?
How are the legend labels stacked?
In this picture, where is the legend?
Which item accounts for the largest proportion?
What is the difference between the maximum value and the minimum value represented by this calendar chart?
What is the maximum value represented by this chart?
Which item has the smallest proportion?
What is the minimum value represented by this calendar graph?
How many points are greater than the average for a certain item?
What is the average of a certain item?
…

Figure 8: Part of the template questions, covering most types of tasks.

Question: what lighting category has the smallest proportion within this segmentation

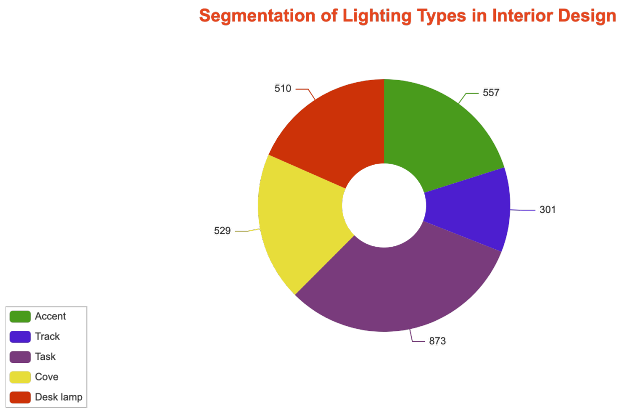
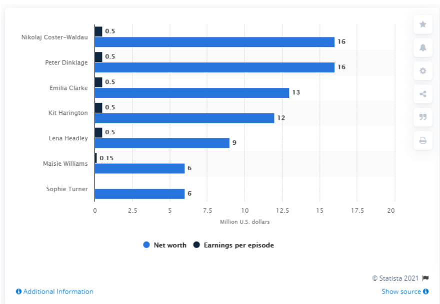


Chart LLaMA: Kitchen lighting

Omni-Chart: The tabular data for this chart is 'Entity, Value
Accent, 557
Track, 301
Task, 873
Cove, 529
Desk lamp, 510'
The question asks what lighting category has the smallest proportion within this segmentation. According to the CSV data we extracted, we know that Track has the smallest proportion with a value of 301, so the answer is Track

Figure 9: results on chart QA task



Omni-Chart

Characteristic, Net worth, Earnings per episode
Nikolaj Coster-Waldau, 16, 0.5
Peter Dinklage, 16, 0.5
Emilia Clarke, 13, 0.5
Kit Harington, 12, 0.5
Lena Headey, 9, 0.5
Maisie Williams, 6, 0.15
Sophie Turner, 6, -

Chart LLaMA

Characteristic, Net worth, earnings per episode
Nicky Wilder,6.5,6
Peter O'Toole,5.5,6
Emilia Clarke,13,13
Kirsten Dunst,12,12
Lauren Bacall,9.5,9
Meryl Streep,8,8
Sophie Turner,6,6
Meryl Streep,6,6
Julianne Moore,5.5,5.5

Figure 10: results on Chart-to-Table task