ChartAssistant: A Universal Chart Multimodal Language Model via Chart-to-Table Pre-training and Multitask Instruction Tuning

Fanqing Meng[2,1], **Wenqi Shao**[1†], **Quanfeng Lu**[1,4], **Peng Gao**[1]
Kaipeng Zhang[1], Yu Qiao[1], Ping Luo[1,3†]

[1]OpenGVLab, Shanghai AI Laboratory [2]Shanghai Jiao Tong University
[3]The University of Hong Kong [4]Nanjing University

Abstract

Charts play a vital role in data visualization, understanding data patterns, and informed decision-making. However, their unique combination of graphical elements (e.g., bars, lines) and textual components (e.g., labels, legends) poses challenges for general-purpose multimodal models. While vision-language models trained on chart data excel in comprehension, they struggle with generalization. To address these challenges, we propose ChartAssistant, a chart-based vision-language model for universal chart comprehension and reasoning. ChartAssistant leverages ChartSFT, a comprehensive dataset covering diverse chart-related tasks with basic (e.g. bars and pies) and specialized (e.g. radars, and bubbles) chart types. It undergoes a two-stage training process, starting with pre-training on chart-to-table parsing to align chart and text, followed by multitask instruction-following fine-tuning. This approach enables ChartAssistant to achieve competitive performance across various chart tasks. Experimental results demonstrate significant performance gains over the state-of-the-art UniChart and ChartLlama methods, especially outperforming them on real-world chart data with zero-shot setting. The code and data are available at https://github.com/OpenGVLab/ChartAst .

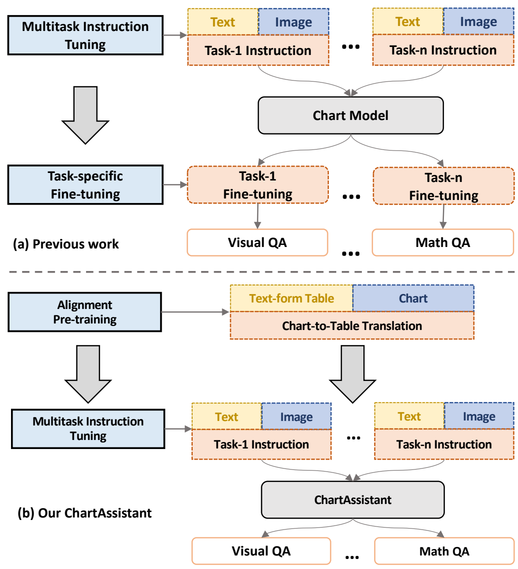


Figure 1: A comparison between previous chart-based models and our proposed ChartAssistant. ChartAssistant first aligns the chart and the text by pre-training on the chart-to-table translation task. After performing multitask instruction tuning, it can solve various downstream tasks.

1 Introduction

People around the world generate a multitude of charts daily, including data visualizations for business reports, market analysis, scientific experiments, and data-driven presentations (Horn, 1998; Hoque et al., 2017, 2022). Charts are an effective tool for understanding data patterns, such as the distributional properties depicted in histograms and growth trends illustrated in line graphs. Developing chart learning methods enables the design of machine analysts with enhanced capabilities to solve various chart-related downstream tasks such as chart question answering (QA) (Masry et al., 2022; Kantharaj et al., 2022a; Methani et al., 2020) and chart summarization (Hsu et al., 2021; Rahman et al., 2022).

However, chart comprehension is challenging due to the intricate visual marks (*e.g.* lines, bars and symbols), implicit numerical information, and complex spatial relationships between elements (*e.g.* axes and labels). Interpreting charts requires specialized knowledge, spatial reasoning, and numerical understanding. The advanced general-purpose multimodal models (Zhang et al., 2023b; Li et al., 2023a; Zhang et al., 2023a) such as LLaVA (Liu et al., 2023b), trained on natural images, struggle with chart-related tasks due to the specific complex-

† Corresponding Authors: shaowenqi@pjlab.org.cn; pluo@cs.hku.edu
This work was done when Fanqing Meng and Quanfeng Lu were interning at Shanghai AI Laboratory.

Findings of the Association for Computational Linguistics: ACL 2024, pages 7775–7803
August 11-16, 2024

ities and relationships unique to charts. Although recent multimodal literate models (Lv et al., 2023; Lee et al., 2023) have achieved impressive results in processing various document-level tasks, they still face difficulties in accurately answering chart-related questions.

In pursuit of universal chart reasoning and comprehension, prior works propose pre-training vision-language models on chart-related tasks as shown in Fig.1(a). For example, both MatCha (Liu et al., 2022b) and UniChart (Masry et al., 2023) perform multitask instructional tuning on chart data. Although these methods exhibit good performance on several chart-related tasks, they require task-specific fine-tuning. Moreover, the existing training data (Methani et al., 2020; Masry et al., 2022) is deficient in image-text annotations aimed at improving the model's comprehension of visual elements and mathematical reasoning, as well as annotated data from the specialized chart types such as box-plots. Due to the above factors, existing chart-based models have poor generalization on various downstream tasks as illustrated in Fig.1(a).

To address these challenges, we propose **ChartAssistant**, a new multimodal model for universal chart comprehension and reasoning. To improve generalization, ChartAssistant is trained on a large-scale chart-specific instruction-tuning benchmark dubbed ChartSFT. The training process involves a two-stage pre-training pipeline which employs chart-to-table pre-training to align the chart and its structured text and then perform joint tuning on multiple chart-related tasks as shown in Fig.1(b). As a result, our ChartAssistant can achieve good results on various chart-related tasks with a single model. We implement ChartAssistant with two variants, *i.e.* ChartAst-D and ChartAst-S. ChartAst-D is built upon Donut (Kim et al., 2021), a lightweight (260M parameters) but powerful vision-language model for visual document understanding. While ChartAst-S is built upon SPHINX (Lin et al., 2023), a large (13B parameters) vision-language model for universal multimodal comprehension. Inherited from SPHINX, our ChartAst-S obtains enhanced chart representation by dynamic resolution processing and mixed visual encoders. Therefore, ChartAst-S offers increased robustness and usability for chart understanding, demonstrating strong performance in various chart-related tasks.

Specifically, we first construct ChartSFT by collecting instruction-following data from various chart-related tasks. To address the limitations of existing chart-based benchmarks (Methani et al., 2020; Masry et al., 2022; Kantharaj et al., 2022a), we introduce several modifications to improve the quality of data annotation: 1) instruction-following data involving various topics for chart-to-table translation is added, which we find helps align the chart and the associated structured text; 2) the chain-of-thought annotations for chart numerical QA task are generated to improve mathematical reasoning abilities (Wei et al., 2022); 3) the task of chart referring question answering is created to enhance the understanding of visual elements and their relationships (Chen et al., 2023; Yang et al., 2023); 4) chart with specialized types such as radar and box plot are included to improve the generalization. Overall, ChartSFT encompasses a larger corpus of instruction-following data, incorporates a wider range of chart-related tasks and types, and features more comprehensive data annotations compared to previous benchmarks (Masry et al., 2022; Methani et al., 2020; Kantharaj et al., 2022a).

Before conducting multitask instruction tuning, as done in existing research (Masry et al., 2023; Liu et al., 2022b), we start with pre-training ChartAssistant on the chart-to-table translation task as shown in Fig.1(b). This task involves parsing a chart and generating a Markdown table. It shares similarities with dense captioning for natural images, allowing the model to interpret the elements and relationships within the chart. Similar to the role of image captioning in training multimodal models (Liu et al., 2023b; Shao et al., 2023; Xu et al., 2023), chart-to-table translation facilitates alignment between the chart and its structured text. Following pre-training, we proceed with multitask instruction tuning using ChartSFT. This two-stage training approach enables ChartAssistant (a single model) to achieve strong performance across a range of chart-related tasks.

The contributions of this paper can be summarized as follows. 1) We present ChartAssistant, a vision-language model for chart comprehension and reasoning. ChartAssistant is versatile enough to solve various chart-related tasks across a wide range of chart types. 2) We build a chart-specific visual instruction-following benchmark dubbed ChartSFT. ChartSFT surpasses existing chart-based benchmarks with its larger instruction-following data corpus, a broader range of tasks and chart types, and more comprehensive data annotations. 3) Extensive experimental results on various downstream tasks demonstrate that ChartAssistant surpasses the previous SoTA method UniChart (Masry et al., 2023) by 50.0%, 28.1% performance gain on numerical QA and ChartQA, respectively. Notably, ChartAssistant continues to significantly outperform existing chart-specific models in the zero-shot setting, with 29.5% performance gain on Real-CQA (Ahmed et al., 2023) compared with Unichart and 23.6% performance gain on ChartLLM (Ko et al., 2023) compared with ChartLlama (Han et al., 2023).

2 Related Work

2.1 Multimodal Foundation Model

Multimodal foundation models (Li et al., 2023a; Zhu et al., 2023) mainly focus on natural images, which have shown remarkable progress, advancing in areas like image captioning (Vinyals et al., 2015) and visual question answering (Vinyals et al., 2015; Johnson et al., 2017). SPHINX (Lin et al., 2023) leverages LLM and multiple visual encoders to achieve advanced performance on multiple multi-modal tasks. Among these, visual document understanding is a topic of both industrial importance and research challenge. Donut (Kim et al., 2021) proposed an OCR-free Transformer trained in end-to-end manner,which is a powerful document understanding model. Nougat (Blecher et al., 2023) is fine-tuned on Donut and useful for academic documents understanding. However, extracting information from real-world images like charts and plots presents unique challenges as compared to natural images or documents. Furthermore, the complexity of queries increases, often involving sophisticated mathematical calculations. As a result, contemporary document models and multimodal foundation models often fall short when tasked with handling chart-related tasks, demonstrating a significant decline in performance (Liu et al., 2022b).

2.2 Chart-specific Vision-Language Model

Some methods modify vision-language models for chart-related tasks (Han et al., 2023; Liu et al., 2023a) or develop plugin for LLM to understand the chart (Xia et al., 2023). MatCha (Liu et al., 2022b) extends Pix2Struct (Lee et al., 2023) by integrating mathematical reasoning and chart data extraction tasks, excelling at chart question answering and chart summarization. Unichart (Masry et al., 2023) and ChartLlama (Han et al., 2023) undergoes multitask instruction tuning on many chart-related tasks, establishing itself as the most versatile and effective chart vision-language model currently available. However, these methods have poor generalization. Furthermore, they struggle with mathematical computations in charts and perform poorly on uncommon chart types such as radars and bubbles. Therefore, we propose ChartSFT, the most extensive dataset to date, supporting a wide variety of chart tasks and types. We develop ChartAssistant using ChartSFT with a two-stage training strategy, capable of handling diverse chart-related tasks.

3 ChartSFT

We construct a large-scale chart-specific instruction-tuning benchmark called ChartSFT by collecting data from various tasks. The composition of ChartSFT is shown in Table 7, as extensively described below. Our ChartSFT consists of 39M pieces of chart-text annotated data, 4.75 and 5.62 times larger than MatCha (Liu et al., 2022b) and UniChart (Masry et al., 2023), respectively, as illustrated in Fig.3. ChartSFT contains charts with both base and specialized types, as presented in Sec. 3.1 and Sec. 3.2, respectively.

Overall, our ChartSFT encompasses nine types of charts by collecting data from various sources as shown in table 12. First, most charts with base types including bar, line, dot-line, and pie are collected from several existing datasets (Masry et al., 2022; Methani et al., 2020; Kantharaj et al., 2022a; Rahman et al., 2022; Li and Tajbakhsh, 2023; Tang et al., 2023; Kantharaj et al., 2022b). Second, we also generate some charts with base types from arXiv tables (arX) and data augmentation techniques (*e.g.* various APIs and figure parameters). In particular, we use ChatGPT to suggest the proper chart type given each table data from arXiv. Third, we synthesize table data which is appropriate for depicting charts with specialized types.

3.1 Chart with Base Types

We collect instruction-following data with base chart types (*i.e.* bars, lines, dot-lines, and pies) from 5 chart-rated tasks, including chart-to-table translation, chart numerical QA, chart referring QA, chart open-ended QA, and chart summarization as shown in Fig.2. Instead of directly utilizing existing chart-based benchmarks, we introduce several modifications to improve the data annotation quality. For each task, we present the details of

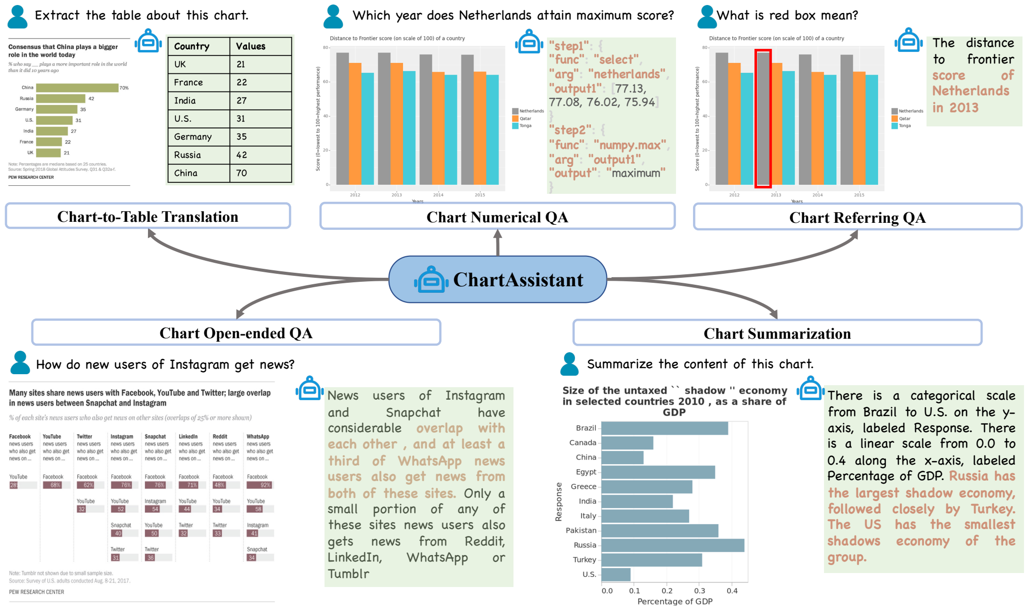


Figure 2: ChartAssistant is pre-trained on various chart-related tasks, and can adeptly perform a wide range of chart-related tasks including chart-to-table translation, numerical QA, referring QA, open-ended QA and chart summarization.

data collection as follows.

3.1.1 Chart-to-Table Translation

The task of chart-to-table translation aims at parsing a chart into its underlying data table in text form. Pre-training with chart-to-table translation enables our ChartAssistant to comprehend the chart's elements and their relationships, facilitating alignment of the chart and its underlying structured text.

Data Collection. We collect 17141 and 224386 pieces of chart-text data from ChartQA and PlotQA for chart-to-table translation. However, these benchmarks vary little in chart styles and involve limited topics. We propose two strategies to address the issue. *i) More Chart Styles.* We re-plot the chart with diverse visualization tools for tables in ChartQA and PlotQA. Specifically, we utilize 5 APIs in Python, including ggplot, plotly, matplotlib, seaborn, and pyecharts, along with over 20 variations in parameters color, size, font type, background, and more. After style augmentation, 220050 pieces of chart-text data are created for chart-to-table translation from PlotQA, respectively. *ii) Table from arXiv Papers.* We collect more real table data to increase the topic diversity. To this end, we crawl 1301932 papers involving various topics such as computer science, biology, finance, and more from arXiv platform (arX). For each paper, we extract the table from the source LaTeX code where table data can be localized in the table environment. We employ ChatGPT (Ouyang et al., 2022) to transform the latex table into the markdown table. We also make the chart in a specific base type (*e.g.* pies) by following ChatGPT's suggestion. We find that ChatGPT works well to generate text in the target format and give appropriate advice for chart types. There are 132719 pieces of chart-text data obtained from the arXiv.

3.1.2 Chart Numerical Question Answering

Chart numerical QA targets at responding to the request about mathematical reasoning given a chart. It requires an accurate understanding of the chart, as well as reasoning and math calculation abilities.

Data Collection. The data for numerical QA mainly comes from the PlotQA benchmark. However, PlotQA generates numerical QA data from 40 templates with limited types of questions and direct final answers, resulting in poor generalization and math reasoning. with our proposed two strategies to improve the data quality below, more than 24M QA pairs are collected. *i) More Templates.* We create 101 templates to generate numerical QA questions automatically involving various types of questions with complex calculations. Here is one template for analyzing the correlation between two items: 'Across all <plural form of X label>, are the <Y label> values of <legend label1> and <legend label2> negatively correlated?' The comparison between templates in our ChartAssistant and PlotQA is provided in Table 1 where we can see that our improved templates encompass larger

Table 1: Comparison of templates for numerical QA between PlotQA and our ChartSFT. 'Num.' denotes the number of templates. We use 'Len.', 'COT Steps' and 'Fun.' to denote the average token length, the number of steps in COT annotation, and the number of functions are needed to obtain the final answer, respectively. Besides templates in PlotQA, ChartSFT newly created 61 templates for numerical QA with higher complexity.

	Num.	Len.	COT Steps	Func.
PlotQA	40	32.83	3.48	2.95
ChartSFT	61 (101)	39.54	5.02	3.90

token lengths and more complex calculations. We present all templates in the Appendix A. *ii) Chain-of-Though (COT) Annotations.* Instead of utilizing the final answer as the response annotation, we generate COT annotation for the final answer, which has been proven to improve the model's mathematical reasoning ability (Wei et al., 2022). We first define a set of available functions to segment the problem's solution into smaller steps, each encompassing function calls and parameters. These steps are then organized into a JSON-formatted text. As shown in Fig.2, the maximum extraction problem is decomposed into a step of data retrieval and a step of maximum calculation. When computing the answers, the backend executes the calculations by following the ordered function calls within the text. This approach not only enhances reasoning ability but also mitigates calculation errors.

3.1.3 Chart Referring Question Answering

We create a new task for chart named referring question answering, considering that users may utilize a set of marks to denote some pieces to their interest in the chart as shown in Fig.2. Note that referring question answering with a bounding box has been explored in general-purpose multimodal models such as GPT4ROI (Zhang et al., 2023c) and Shikra (Chen et al., 2023) where the referential QA has been shown to benefit comprehending spatial relationships. The task of referring QA is expected to enhance the understanding of visual elements and their relationship in the chart.

Data Collection. We extend a part of COT annotations for numerical QA in Sec.3.1.2 to the task of Referring QA. Three steps are conducted to produce referring QA pairs with diverse patterns. i) The color, size, and width are randomly selected to make the mark. ii) We use several marks such as an arrow and a bounding box to refer to an item in the chart. iii) Multiple marks can be depicted in the same chart to describe the relationships between

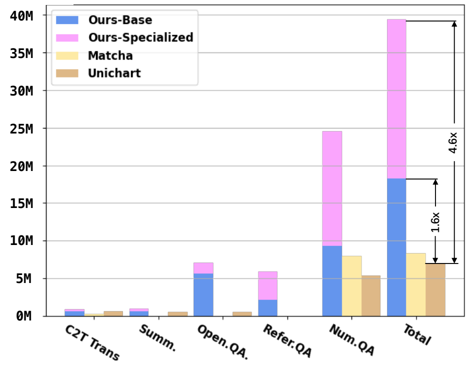


Figure 3: Comparison between ChartSFT and datasets from previous methods. Our dataset surpasses the best previous dataset in UniChart (Masry et al., 2023) by 4.6 times in total and supports a greater variety of chart tasks and types.

elements. Overall, we collect 5899842 pieces of data for the chart referring QA.

3.1.4 Chart Open-ended QA

Chart open-ended QA (OpenQA) deals with open-ended questions regarding charts as illustrated in Fig.2. It requires both low-level Chart comprehension and high-level reasoning abilities.

Data Collection. We collect data from existing benchmarks, such as plotQA (Methani et al., 2020), ChartQA (Masry et al., 2022), OpenCQA (Kantharaj et al., 2022a) and ScigraphQA (Li and Tajbakhsh, 2023). We further introduce our collected table data from arXiv in Sec.3.1.1 for this task. *i) Open-ended QA data by ChatGPT.* Other than tabular data crawled in Sec.3.1.1, we extract corresponding captions, and the first paragraph describing the table from the source code of the paper. By utilizing ChatGPT, we generate 3 open-ended QA pairs for each table by feeding the table and the descriptive information.

By putting the above benchmarks together, our ChartSFT covers diverse topics for Open-ended QA. In total, there are 7075243 pieces of data for this task.

3.1.5 Chart Summarization

Chart Summarization is a vital task aimed at generating concise and informative summaries for various types of charts, which has been studied extensively (Herdade et al., 2019; Tang et al., 2023; Kantharaj et al., 2022b).

Data Collection. We collected a substantial amount of existing open-source datasets (Tang et al., 2023; Kantharaj et al., 2022b; Rahman et al., 2022; Kantharaj et al., 2022a), but the scale is still not sufficient. Therefore, we further incorporate

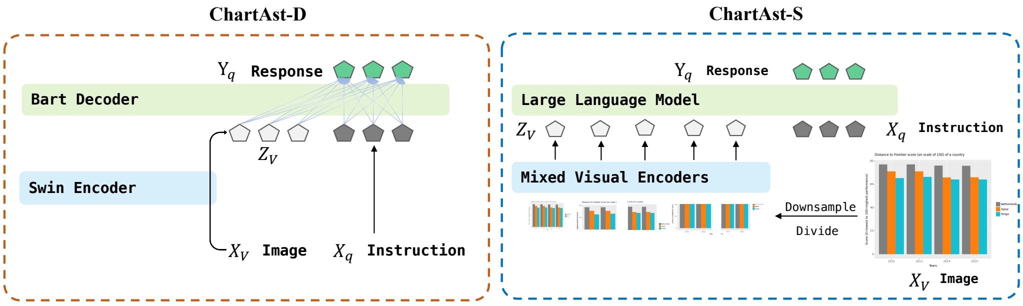


Figure 4: Illustration of ChartAst-D adopted from Donut (Kim et al., 2021) and ChartAst-S inherited from SPHINX (Lin et al., 2023).

a large-scale chart summarization dataset generated through Knowledge Distillation by Unichart (Masry et al., 2023) into our training process. There are 1006738 pieces of data for the chart summarization task.

3.2 Charts with Specialized Types

Previous chart-based models have exhibited poor performance when dealing with specialized chart types, such as radar, area, histogram, bubble, and box-plot. To enhance the model's generalization capabilities, we have trained our ChartAssistant on these charts with specialized types. To overcome the challenge of obtaining large-scale real-world chart data, we have employed synthetic data generation techniques. For more detailed information, please refer to Appendix A Through this approach, we can obtain a substantial and diverse collection of complex charts across these specialized types.

4 Our ChartAssistant

4.1 Architecture

The key to completing the tasks related to charts lies in accurately understanding the content of the charts. As shown in Fig. 4, we implement ChartAssistant with two variants, *i.e.* ChartAst-D and ChartAst-S, which have 260M and 13B parameters in total. In addition, their input image resolutions are 224×224 and 448×448, respectively. Both ChartAst-D and ChartAst-S perform well in many chart-related tasks. But ChartAst-D has a smaller size and ChartAst-S enjoys better generalization.

ChartAst-D is a vision-language model for chart understanding built upon Donut (Kim et al., 2022). It consists of a visual encoder Swin-Base (Liu et al., 2021) and a textual BART decoder (Lewis et al., 2019). For an input image X_V, the visual encoder employs fixed-sized non-overlapping windows to divide the image and performs self-attention layers to consolidate information across these windows, which transforms the image into a set of tokens $Z_V = \left\{\mathbf{z}_i \mid \mathbf{z}_i \in R^d, 1 \leq i \leq n\right\}$, where n is encoded token length and d is the token size. By taking Z_V as key and value and tokens of text instruction X_q as the query, the BART decoder generates the corresponding response $Y_q = (\mathbf{y}_i)_{i=1}^m$, and m is the length of responses.

ChartAst-S is a large vision-language model for chart understanding built upon SPHINX (Lin et al., 2023). For high-resolution images, it preserves the original information through sampling and partitioning methods, ensuring greater fidelity to the image content. Moreover, SPHINX leverages the abundant prior knowledge of LLM (Touvron et al., 2023) to handle various tasks such as visual question answering and image summarization. Specifically, for an input image X_V. ChartAst-S incorporates multiple visual encoders to extract more informative visual features Z_V, such as DINOv2 (Oquab et al., 2023), CLIP (Radford et al., 2021), and ConvNeXt (Woo et al., 2023). Unlike ChartAst-D where visual tokens are involved in a language decoder with a cross-attention module, ChartAst-S directly appends visual tokens to the text tokens X_q. The merged tokens are then fed into the LLM to generate the response. Thanks to the intricate design of the visual encoder and the powerful reasoning ability of LLM, ChartAst-D generalizes well in various real-world chart-related applications.

4.2 Training

In our ChartSFT, we have a corresponding instruction X_q and response Y_q for each image X_V. We input these image-text pairs into the model. The objective is to minimize the cross-entropy loss of predicting the next token. To improve the generalization in various downstream tasks, we adopt a two-stage training pipeline to train our ChartAst-D and ChartAst-S below.

Stage I: Pretraining on Chart-to-table Translation. Charts are special images that visualize the data and underlying relationships between elements in the chart. Understanding the numerical values and their meanings is a prerequisite for completing downstream tasks related to charts. Given a chart X_V^{c2t}, this stage is to convert the chart into a text-form table Y_q^{c2t} under the instruction X_q^{c2t}. Here the superscript $c2t$ indicates the instruction-following data comes from the task of chart-to-table translation. Our training loss function for Stage I is given by $\mathcal{L}^{\text{Stage1}} = -\sum_{i=1}^{m} \log P_\theta(Y_{q,i}^{c2t}|X_V^{c2t}, X_q^{c2t}, Y_{q,<i}^{c2t})$ where $Y_{q,<i}^{c2t}$ are all the response tokens before the current prediction token $Y_{q,i}^{c2t}$. θ are the learnable weights initialized from the pre-trained weights of the Donut model (Kim et al., 2021). By the pre-training, we align the chart with its structured text-form table, enabling the model to comprehend elements in charts and their relationships. We show that this strategy better serves the multitask instruction tuning in Sec.5.4.

Stage II: Multitask Instruction Tuning. In this stage, we put all the instruction-following data together from five tasks in our ChartSFT. We employ a single model to solve all the tasks. Our training loss function for Stage II is given by $\mathcal{L}^{\text{Stage2}} = -\sum_{k\in\Omega}\sum_{i=1}^{m} \log P_\theta(Y_{q,i}^{k}|X_V^{k}, X_q^{k}, Y_{q,<i}^{k})$, where Ω is the set of instruction-following data from all tasks in ChartSFT and θ are the learnable weights initialized from the checkpoint in the Stage I. During training, we sample the data from each task with certain proportions as provided in our experimental setup in Appendix B. By multitask instructional tuning, our ChartAssistant exhibits strong performance on all the tasks.

5 Experiment

we present our experimental setup in Appendix B, where we indicate the training details. After that, we provide an overview of the selected baselines and evaluation details in Sec.5.1 and demonstrate the superior effectiveness of our method through extensive experiments in Sec.5.2 .

5.1 Baselines and Evaluation

Evaluation. We assess the performance of ChartAssistant across various tasks and datasets. Following the evaluation of Unichart (Masry et al., 2023), we utilize Chart-to-text (Kantharaj et al., 2022b) for evaluating chart summarization task, OpenCQA (Kantharaj et al., 2022a) and ChartQA (Masry et al., 2022) for open-ended question answering task. To evaluate numerical question answering and referring question answering, we sample test sets from the datasets constructed in Sec.3.1.2 and Sec.3.1.3 called MathQA and ReferQA. Lastly, we conduct separate evaluations on base type and specialized type charts to highlight the superior performance of our method more explicitly. We put a detailed description of the dataset and more experiments in Appendix B.

Table 2: A comparison of the results of ChartAssistant with the existing Chart model on five tasks with base type charts, which shows that ChartAssistant is ahead of the rest of the models on all tasks. Bold indicates best results, italics indicate that the model is not trained on this task.

		ChartQA		Chart-to-Text		Chart-to-Table			
Model	Size	aug.	human	Pew	Statista	ChartQA	OpenCQA	MathQA	ReferQA
T5	223M	41.0	25.1	10.5	35.3	-	9.3	-	-
Chart-T5	400M	74.4	31.8	9.10	37.5	-	-	-	-
Donut	260M	78.1	29.8	7.2	38.2	87.4	13.1	36.3	*6.2*
Pix2Struct	300M	81.6	30.5	10.3	38.0	85.9	12.7	35.6	*5.8*
Monkey	x	84.6	44.6	*0.4*	*1.7*	*0.6*	*11.3*	*5.7* SPHINX	13B
11.3	*21.7*	*3.2*	*4.1*	*9.4*	*5.9*	*4.4*	*7.2*		
Qwen	9.6B	78.9	44.3	*0.5*	*2.6*	-	*1.3*	*4.8*	*4.9*
Blip2	4B	*1.4*	*7.8*	*0.2*	*0.8*	-	*1.7*	*6.4*	*0.4*
MatCha	300M	88.9	38.8	12.2	39.4	89.6	*6.5*	57.8	*8.3*
Unichart	260M	87.8	43.9	12.5	38.1	91.1	14.8	23.9	*11.9*
ChartLlama	13B	90.4	48.9	14.2	40.7	90.0	*4.7*	*5.8*	*9.9*
ChartAst-D	260M	91.3	45.3	14.0	40.2	**92.0**	14.9	72.1	64.2
ChartAst-S	13B	**93.9**	**65.9**	**15.2**	**41.0**	91.6	**15.5**	**73.9**	**67.9**

Baselines. We choose SPHINX (Lin et al., 2023), Monkey (Li et al., 2023b), Blip2-flant5-xl (Li et al., 2023a), Qwen-VL (Bai et al., 2023), ChartLlama (Han et al., 2023), Unichart (Masry et al., 2023), MatCha (Liu et al., 2022b), Pix2Struct (Lee et al., 2023), T5 (Raffel et al., 2020) and Chart-T5 (Zhou et al., 2023) as baselines. We provide a detailed description in Appendix B.

5.2 Main Results

Charts with base types. In table 2, we present a comprehensive summary of ChartAssistant's performance on charts with base types across chart-related tasks. It demonstrates that ChartAssistant consistently outperforms the baseline across all tasks. In particular, we surpass the current leading methods ChartLlama by 17% and 2.5% on ChartQA-human and ChartQA-augment, respectively. Besides, Most existing models struggle with numerical question answering, while the COT answer significantly enhances performance for our models, demonstrating a substantial 16.1% improvement over MatCha. Notably, existing models cannot handle the chart referring question answering task effectively. Overall, our model is the top performer across all chart-related tasks. It is important to note that both the performance of Unichart and MatCha's are obtained after task-specific fine-

Table 3: In comparison with other chart-related multimodal models in a zero-shot setting, ChartAssistant-S significantly outperforms existing models across all tasks in the zero-shot scenario.

	RealQA			
Model	Math	Extract	ChartLLM	StructChart
Unichart	13.0	33.0	11	41.5
MatCha	16.0	27.5	11	23.3
ChartLlama	10.0	13.0	55	38.3
Monkey	6.0	7.5	32	-
GeminiPro	11.5	12.0	66	37.5
GPT-4V	6.0	20.0	**81**	11.6
ChartAst-D	15.0	36.0	13	39.4
ChartAst-S	**32.0**	**43.5**	68	**45.3**

tuning with the training set of the test dataset, whereas ChartAssistant's results are obtained using a single model after a two-stage training.

5.3 Zero-shot Study

To validate the generalization of ChartAssistant, we test it on samples not included in the training set. To this end, We sample 200 examples from StructChart (Xia et al., 2023) and RealCQA (Ahmed et al., 2023), including two types: mathematical computation and numerical extraction, and collect all 48 publicly available examples from ChartLLM (Ko et al., 2023) for tasks like chart-to-table translation, chart-based question answering, and summarization. For evaluation, RealCQA uses accuracy within a 5% error margin, ChartLLM employs GPT-4 scoring used in ChartLlama (Han et al., 2023), while StructChart is evaluated using RMS_{F_1} metrics. As shown in table 3, we find ChartLlama performs poorly in precise numerical question answering but excels in summarization tasks. We attribute this to the robust language capabilities of LLM. But ChartAssistant surpasses existing models in tasks such as precise numerical question answering in OCR and summarization, which involves generating long texts. Furthermore, we observe that if the model's decoder is not powerful enough, errors are more likely to occur in the zero-shot setting when tasked with generating long text outputs, such as in summarization or providing answers in COT format. The use of Large Language Models (LLMs) can significantly alleviate this issue. Overall, our ChartAst-S exhibits the best zero-shot performance across all tasks.

5.4 Ablation Study

We thoroughly analyze the key aspects of our approach. We first consider the significance of alignment pre-training and the referring question answering task. Furthermore, We put more experiments in Appendix C, including the key component of ChartSFT. We adopt ChartAst-D to illustrate the superiority of our designed ChartSFT, as well as to emphasize the importance of the training strategy.

Table 4: A comparison of the results of ChartAssistant with its variants on five tasks with base type charts, which indicates that the alignment pretraining and the referring question answering task play a crucial role in enhancing the overall performance.

	ChartQA		Chart-to-Text		Chart-to-Table			
Model	aug.	human	Pew	Statista	ChartQA	OpenCQA	MathQA	ReferQA
Ours-D w/o align	89.0	42.1	13.7	38.3	89.5	14.3	62.3	60.1
Ours-D w/o refer	89.2	41.2	14.0	38.6	90.7	14.6	60.2	-
Ours-D	**91.3**	**45.3**	**14.0**	**40.2**	**92.0**	**14.9**	**72.1**	**64.2**

The impact of alignment pretraining. We initially validate the importance of alignment pretraining. We ensure that the "Ours w/o align" version of the model is trained for the same number of iterations as the full ChartAssistant model. Table 4 shows that using only multitask instruction tuning falls considerably behind two-stage training strategies. Exact numerical recognition greatly influences mathematical calculation accuracy, leading to a 9.8% and 3.2% performance drop for MathQA and ChartQA-human tasks. We think alignment pre-training, which allows the model to learn chart-table correlations, helps the model better adapt during multitask instruction tuning than handling these processes separately (Liu et al., 2023b).

The impact of referring question answering task. In our experiments, we have observed that integrating referring question answering into multitask instruction tuning training can enhance the model's performance in other tasks. As shown in table 4, incorporating the referring question answering task leads to improvements across almost all tasks, particularly in tasks requiring mathematical reasoning. For instance, the average performance in ChartQA improves by 3.1%, and in MathQA, it improves by 11.9%. We believe that this task strengthens the model's ability to understand the visual elements and their relationship in the chart, which contributing to overall performance enhancement (Zhang et al., 2023c; Chen et al., 2023).

Error case analysis. Although our model currently achieves the most competitive performance on various chart-related tasks, there is still significant room for improvement. For example, ChartAst-S achieves only 65.9% accuracy on the human split of ChartQA. As an example, we conduct an error analysis on this segment. Within ChartQA-human, there are a total of 1,250 QA

Table 5: Error statistics of ChartAssistant on ChartQA-human.

Error	Generating CoT	Computation	Extracting numbers	Other
ChartAst-S	230	163	11	22

pairs. The questions mainly focus on extracting elements and solving mathematical questions and also include some questions about basic chart attributes. For ChartAst-S, there are a total of 426 errors.

As shown in table 5, we categorize the errors into i) inability to generate CoT; ii) generated CoT but with computation errors; iii) errors in extracting numbers; and iv) others. The results are reported in Table D where we see that ChartAst-S mainly occur in mathematical calculations. Error types i) and iii) account for less than 8% of the total errors. Among mathematical questions, 58.5% of errors result from wrong COTs. Although the remaining 41.5% of answers contain the correct COT steps, the calculation error occurs because the extracted elements are wrong. Hence, our future work would focus on enhancing the model's ability to extract multiple elements based on the question.

Further, we analyze the errors related to generating the wrong CoT or failing to generate CoT. Approximately 44% of these errors occur because the question incorporates visual information from the chart, such as size, colour, or position (e.g., "What is the value of the longest blue bar?"). Previous works like TinyLVLM-eHub (Shao et al., 2023) also found that multimodal models are deficient in identifying visual commonsense such as shape and colour. It implies that the ability to recognize visual commonsense should be improved.

6 Conclusion

Our work is aimed at developing a generalized multimodal model for chart-related tasks. We propose ChartSFT, a comprehensive and expansive dataset with the most diverse range of supported chart tasks and types. In conjunction, we suggest ChartAssistant, a multimodal model trained using a two-stage strategy over ChartSFT, which can achieve state-of-the-art results across multiple chart-related downstream tasks. Through detailed experiments, we further demonstrate the superiority of it.

7 Limitations

Due to limitations in the training data and the large vision-language model employed, the current version of ChartAssistant performs significantly better on English than on those in other languages. To overcome this issue, we aim to enhance our model by incorporating multilingual training data and expanding the range of chart types supported.

References

Arxiv. https://arxiv.org/.

Saleem Ahmed, Bhavin Jawade, Shubham Pandey, Srirangaraj Setlur, and Venu Govindaraju. 2023. Realcqa: Scientific chart question answering as a test-bed for first-order logic. In *International Conference on Document Analysis and Recognition*, pages 66–83. Springer.

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*.

Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. 2023. Nougat: Neural optical understanding for academic documents. *arXiv preprint arXiv:2308.13418*.

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

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*.

Simao Herdade, Armin Kappeler, Kofi Boakye, and Joao Soares. 2019. Image captioning: Transforming objects into words. *Advances in neural information processing systems*, 32.

Enamul Hoque, Parsa Kavehzadeh, and Ahmed Masry. 2022. Chart question answering: State of the art and future directions. In *Computer Graphics Forum*, volume 41, pages 555–572. Wiley Online Library.

Enamul Hoque, Vidya Setlur, Melanie Tory, and Isaac Dykeman. 2017. Applying pragmatics principles for interaction with visual analytics. *IEEE transactions on visualization and computer graphics*, 24(1):309–318.

Robert E Horn. 1998. Visual language. *MacroVu Inc. Washington*.

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

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910.

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*.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. 2022. Ocr-free document understanding transformer. In *European Conference on Computer Vision*, pages 498–517. Springer.

Geewook Kim, Teakgyu Hong, Moonbin Yim, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. 2021. Donut: Document understanding transformer without ocr. *arXiv preprint arXiv:2111.15664*, 7:15.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Hyung-Kwon Ko, Hyeon Jeon, Gwanmo Park, Dae Hyun Kim, Nam Wook Kim, Juho Kim, and Jinwook Seo. 2023. Natural language dataset generation framework for visualizations powered by large language models. *arXiv preprint arXiv:2309.10245*.

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.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

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

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

Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. 2023b. Monkey: Image resolution and text label are important things for large multi-modal models. *arXiv preprint arXiv:2311.06607*.

Ziyi Lin, Chris Liu, Renrui Zhang, Peng Gao, Longtian Qiu, Han Xiao, Han Qiu, Chen Lin, Wenqi Shao, Keqin Chen, et al. 2023. Sphinx: The joint mixing of weights, tasks, and visual embeddings for multi-modal large language models. *arXiv preprint arXiv:2311.07575*.

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, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022.

Tengchao Lv, Yupan Huang, Jingye Chen, Lei Cui, Shuming Ma, Yaoyao Chang, Shaohan Huang, Wenhui Wang, Li Dong, Weiyao Luo, et al. 2023. Kosmos-2.5: A multimodal literate model. *arXiv preprint arXiv:2309.11419*.

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.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

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

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

Raian Rahman, Rizvi Hasan, and Abdullah Al Farhad. 2022. *ChartSumm: A large scale benchmark for Chart to Text Summarization*. Ph.D. thesis, Department of Computer Science and Engineering (CSE), Islamic University of ....

Wenqi Shao, Yutao Hu, Peng Gao, Meng Lei, Kaipeng Zhang, Fanqing Meng, Peng Xu, Siyuan Huang, Hongsheng Li, Yu Qiao, et al. 2023. Tiny lvlm-ehub: Early multimodal experiments with bard. *arXiv preprint arXiv:2308.03729*.

Benny J Tang, Angie Boggust, and Arvind Satyanarayan. 2023. Vistext: A benchmark for semantically rich chart captioning. *arXiv preprint arXiv:2307.05356*.

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

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

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

Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. 2023. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16133–16142.

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*.

Peng Xu, Wenqi Shao, Kaipeng Zhang, Peng Gao, Shuo Liu, Meng Lei, Fanqing Meng, Siyuan Huang, Yu Qiao, and Ping Luo. 2023. Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models. *arXiv preprint arXiv:2306.09265*.

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. 2023. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*.

Ao Zhang, Hao Fei, Yuan Yao, Wei Ji, Li Li, Zhiyuan Liu, and Tat-Seng Chua. 2023a. Transfer visual prompt generator across llms. *arXiv preprint arXiv:2305.01278*.

Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. 2023b. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*.

Shilong Zhang, Peize Sun, Shoufa Chen, Min Xiao, Wenqi Shao, Wenwei Zhang, Kai Chen, and Ping Luo. 2023c. Gpt4roi: Instruction tuning large language model on region-of-interest. *arXiv preprint arXiv:2307.03601*.

Mingyang Zhou, Yi R Fung, Long Chen, Christopher Thomas, Heng Ji, and Shih-Fu Chang. 2023. 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 ChartSFT

A.1 Chart Types

Our ChartBench encompasses nine types of charts by collecting data from various sources as shown in table 6. First, most charts with base types including bar, line, dot-line, and pie are collected from several existing datasets (Masry et al., 2022; Methani et al., 2020; Kantharaj et al., 2022a; Rahman et al., 2022; Li and Tajbakhsh, 2023; Tang et al., 2023; Kantharaj et al., 2022b). Second, we also generate some charts with base types from arxiv tables and data augmentation techniques (*e.g.* various APIs and figure parameters). In particular, we use ChatGPT to suggest the proper chart type given each table data from arxiv. Third, we synthesize table data which is appropriate for depicting charts with specialized types.

A.2 Details of Chart Data Generation in ChartSFT

We illustrate the pipeline of data generation in Fig. 5. In a concrete manner, the chart data are generated in the following stages:

Stage 1: Table generation: Taking into account the diversity of tabular data, we have predefined over 20 types of probability density distributions, including normal distribution, uniform distribution, beta distribution, Laplace distribution, and more. For each sample, we randomly choose one type of probability density distribution and utilize it to generate values. For different types of charts, we impose further constraints on these values based on their characteristics. (e.g., value range, ratio of positive and negative values, range interval). For radar, bubble and area charts, We directly utilize randomly generated values as the tabular data. For histogram and box plot, we generate an array of extensive values using this distribution and calculate the statistical metrics of this array to serve as the tabular data (e.g., frequencies corresponding to histograms, upper whiskers corresponding to box plots). And then we use the generated data to prompt ChatGPT for creating titles, legends, and labels that align with the numerical characteristics.

Stage 2: Chart generation: To ensure the diversity of the generated charts, we utilize multiple plot APIs, such as matplotlib, plotly, pyecharts, ggplot, seaborn, altair, and more, to plot a variety of styles of the chart. For each chart, we randomly select the following parameters: line (style, thickness), font (style, size, bold, italic), colors, markers, the position of the elements (title, labels, legends), the size of the charts and so on. Besides our own synthetic tabular data, we also use the table from PlotQA (Methani et al., 2020), ChartQA (Masry et al., 2022), ChartSumm (Rahman et al., 2022) and Chart-To-Text (Kantharaj et al., 2022b) to plot the charts for area and radar charts.

Stage 3: Instruction Data generation: For the chart summarization and open-ended QA tasks, we instruct ChatGPT to build datasets by supplying both the table and the corresponding types of charts. For numerical QA and referring QA tasks, we adhere to the approach of the chart with base types by crafting a series of mathematical question templates tailored to the distinct characteristics of various chart types. Subsequently, we manually generate answers with COT annotations.

We adopted a flexible approach by combining ChatGPT with human intervention, which included the utilization of predefined distributions and custom coding of plot API, among other techniques. Through this three-stage chart data generation process, we ensured the diversity and complexity of the table, chart, and instruction data, respectively. As a result, we were able to generate a substantial volume of diversified high-quality chart data.

A.3 Numerical QA Templates

We present all the Numerical QA templates in this section. We systematically record both the number of steps in the COT annotation and the number of unique functions used to obtain for each template. Fig 12 shows 101 general templates designed for charts with different types. However, not all of these general templates are applicable to all types of charts. Hence, we've customized templates to match the unique characteristics of several specific chart types, such as box plots, bubbles, histograms, and pies, as demonstrated in fig . 17

A.4 Details of Referring QA in ChartSFT

In this section, We introduce the details of the generation pipeline of referring QA in our ChartSFT.

Chart Generation. We generate charts with the referring box in two ways. 1) For base types of charts, we utilize the bounding box annotations from plotQA to add referring markers onto their original images. 2) For specialized types of charts, we directly generate charts with integrated referring markers leveraging certain Python API(e.g., matplotlib) functionalities. Fig.11 shows different types of charts with different referring markers.

Table 6: Chart type distribution of the multitask instruction tuning, we are not including SciGraphQA (Li and Tajbakhsh, 2023) and ChartSumm (Rahman et al., 2022) because these datasets do not contain information about chart types.

Datasets	Bar	Line	Dot-line	Pie	Area	Hist	Radar	Bubble	Box
ChartQA (Masry et al., 2022)	84.8%	12.2%	0.0%	3.0%	0.0%	0.0%	0.0%	0.0%	0.0%
PlotQA (Methani et al., 2020)	67.0%	16.5%	16.5%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
OpenCQA (Kantharaj et al., 2022a)	71.7%	24.6%	0.6%	3.0%	0.1%	0.0%	0.0%	0.0%	0.0%
Vistext (Tang et al., 2023)	50.1%	24.2%	0.0%	0.0%	25.6%	0.0%	0.0%	0.0%	0.0%
Chart-to-text (Kantharaj et al., 2022b)	82.8%	13.6%	1.6%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
arXiv	71.6%	17.1%	11.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Data Aug.	56.5%	17.0%	11.5%	15.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Synthetic	0.0%	0.0%	0.0%	0.0%	23.1%	25.8%	20.9%	19.1%	11.1%
Total	44.3%	11.3%	8.0%	3.6%	7.8%	8.4%	6.8%	6.2%	3.6%

Table 7: Summary of utilized datasets and data volumes for each task. We use datasets we built ourselves as well as these open source datasets: ChartQA (Masry et al., 2022), PlotQA (Methani et al., 2020), OpenCQA (Kantharaj et al., 2022a), SciGraphQA (Li and Tajbakhsh, 2023), VisText (Tang et al., 2023), Chart-to-Text (Kantharaj et al., 2022b), ChartSumm (Rahman et al., 2022).

ChartQA	PlotQA	OpenCQA	ScigraphQA	Vistext	Chart-to-text	ChartSumm	arXiv	Data Aug.	SpecializedTypes	Total
Chart-to-Table Translation										
17141	224386	0	0	0	0	0	132719	220050	317662	911958
Numerical Question Answering										
0	3997388	0	0	0	0	0	0	5318500	15178693	24494581
Referring Question Answering										
0	0	0	0	0	0	0	0	2139567	3760275	5899842
Open-ended Question Answering										
30219	4362236	7724	659309	0	0	0	408658	128105	1478952	7075203
Chart Summarization										
0	157070	7724	0	12441	44096	84363	0	356248	419895	1006738

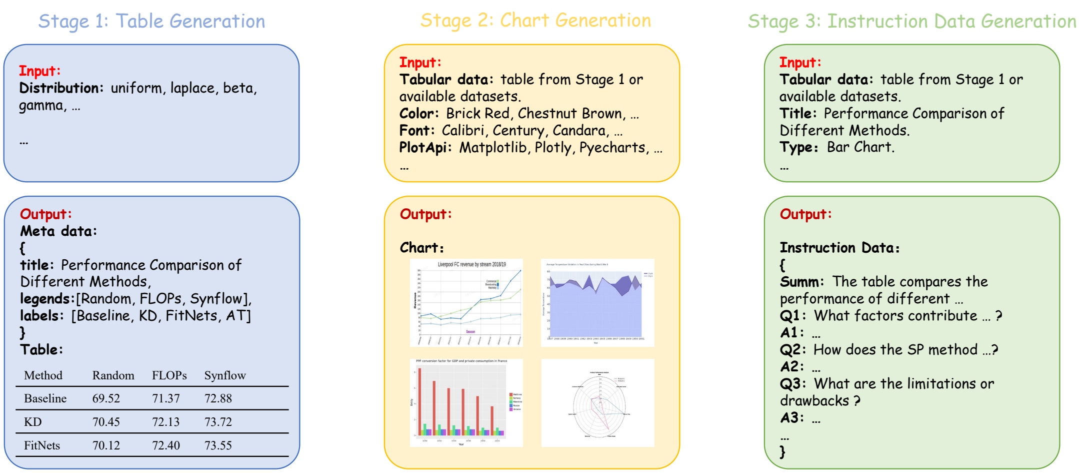


Figure 5: The pipeline of Chart Data Generation in ChartSFT, which consists of three important stages.

QA Generation. Following the pipeline used in generating numerical QA templates, we extend its application to the referring QA task. As outlined in fig. 18, we define a total of 114 templates, encompassing questions related to label recognition and mathematical calculations. Note that the x_tick of line and area charts is continuous, therefore, we tailor these templates to accommodate such scenarios.

B Experiments

B.1 Experimental Setups

We begin by conducting alignment pre-training, utilizing the chart-to-table translation task for 65k steps. Following that, we engage in multitask instruction tuning. We employ the Adam optimizer (Kingma and Ba, 2014) with a scheduled learning rate, where the initial rate is set to 5e-5 for

ChartAst-D and 2e-6 for ChartAst-S. The input resolution is established at 448×448, while the maximum length in the decoder is defined as 1536 for ChartAst-D and 2048 for ChartAst-S. After training for four epochs for ChartAst-D and only one epoch for ChartAst-S, we perform testing on multiple downstream tasks. During inference, each task receives an image and a textual instruction as input, and the model generates a textual answer. All training processes are carried out on 16xA100 80GB GPUs. ChartAst-S outperforms ChartAst-D and has stronger robustness. This is partly due to the special high-resolution image handling method employed by ChartAst-S, which retains more detailed chart information. Additionally, ChartAst-S incorporates richer pre-training knowledge and the larger model possesses greater robustness.

Evaluation. We assess the performance of ChartAssistant across various tasks and datasets. Following the evaluation of Unichart (Masry et al., 2023), we utilize the test set of Chart-to-text (Kantharaj et al., 2022b) for evaluating chart summarization task, and test sets of OpenCQA (Kantharaj et al., 2022a) and ChartQA (Masry et al., 2022) for open-ended question answering task. The ChartQA dataset consists of two subsets: augmented and human. The augmented set comprises machine-generated summaries with a predominantly extractive nature, while the human set contains manually crafted summaries that require more advanced reasoning. The Chart-to-Text task encompasses two sets named "Pew" and "Statista" indicating the origin of the image examples. In the Pew set, summaries are automatically extracted from areas surrounding the images, while in the Statista set, summaries are authored by human annotators. We use ChartQA and PlotQA to evaluate chart-to-table translation tasks due to their various chart styles. To evaluate numerical question answering and referring question answering, we sample test sets from the datasets constructed by ourselves called MathQA and ReferQA. When evaluating the effectiveness, we test three times and take the average.

Metrics. For evaluating ChartQA, MathQA, and ReferQA, we adopt the approach used in previous studies (Liu et al., 2022b; Masry et al., 2023), which considers relaxed correctness (allowing for an exact match with tolerance for a 5% numerical error). As for Chart-to-Text and OpenCQA, we employ BLEU as the evaluation metric following previous works (Liu et al., 2022b; Masry et al., 2023). For chart-to-table translation, we use RMS_{F1} from DePlot (Liu et al., 2022a).

Table 8: The chart-to-table translation performance of ChartAssistant and some baselines on plotQA.

Dataset	ChartAst-S	ChartAst-D	MatCha	Unichart
PlotQA	**95.6**	90.1	82.7	70.8

Baselines. We choose SPHINX (Lin et al., 2023), Blip2-flant5-xl (Li et al., 2023a), Qwen-VL (Bai et al., 2023), ChartLlama (Han et al., 2023), Unichart (Masry et al., 2023), MatCha (Liu et al., 2022b), Pix2Struct (Lee et al., 2023), T5 (Raffel et al., 2020) and Chart-T5 (Zhou et al., 2023) as baselines. ChartLlama and Unichart are the current state-of-the-art models that handles the maximum number of chart tasks and delivers the best overall performance. Besides, Unichart also considers the open-ended QA task. MatCha outperforms previous models in mathematical calculations. Pix2Struct and Donut stands out as an excellent document understanding model. We fine-tune these document models on the train set of the respective evaluation datasets and present the results. T5 is a text-to-text model and needs OCR-based system to extract the data table from the chart image, Chart-T5 is a model modified from T5 for chart-related tasks. We use the results from Unichart (Masry et al., 2023) for them. SPHINX (Lin et al., 2023), Blip2(Li et al., 2023a) and Qwen-VL (Bai et al., 2023) are all commonly used large vision-language models at present. We observe that these models underperform in processing Chart tasks. Finally, ChartLlama, utilizing LLaVA for training on Chart data, demonstrates superior performance in Chart tasks. Therefore, we only compare with ChartLlama.

B.2 More experiments

Specialized type charts. Following the similar training strategy shown in Fig.2, we fine-tune ChartAssistant on chart data of specialized types. As depicted in table 9, compared to the current chart-specific vision-language models, none of them can generalize effectively to specialized types of charts due to lack of these training data. ChartAssistant demonstrates an absolute advantage in all five tasks related to specialized types of charts compared to them.

Chart-to-Table translation on PlotQA A significant portion of the ChartQA (Masry et al., 2022) dataset labels corresponding numerical data on the charts, but there also exists a considerable amount

Table 9: A comparison of the results of ChartAssistant with other chart-specific models on five tasks with specialized type charts. Use BLEU to evaluate summarization and open-ended QAs.

Model	C2T Trans.	Summ.	Open.QA	Num.QA	Refer.QA
MatCha	17.1	6.3	5.1	7.2	-
Unichart	18.4	6.3	5.4	5.9	-
ChartLlama	19.4	9.2	8.4	2.4	-
ChartAst-D	68.3	19.7	25.7	42.5	65.2
ChartAst-S	**75.6**	**22.0**	**27.8**	**49.8**	**68.4**

of charts where the numbers are not visualized. Consequently, we utilize the PlotQA (Methani et al., 2020) dataset to conduct additional chart-to-table translation experiments. As table 8 shows, the results indicate that compared to the ChartQA dataset, the ChartAssistant demonstrates a more significant advantage when implemented on the PlotQA dataset.

C Ablation Study

We thoroughly analyze the key aspects of our approach.

The impact of arXiv data. we conduct experiments by excluding the arXiv data at two distinct stages: the alignment pre-training (stage 1), and the multitask instruction tuning (stage 2). As shown in table 10, it demonstrates that the arXiv dataset significantly assists the model in aligning charts with tables, thereby improving the performance across various tasks. We believe this is due to the fact that in comparison to existing chart-to-table translation datasets, the arXiv dataset boasts more diversity in terms of style and context; Besides, the open-ended question-answering task contributed by the arXiv dataset is proved to be pivotal for the multitask instruction tuning. We note that the removal of this leads to a drop in the performance of all tasks, most notably math QA and the referring QA. The possible reason for this is because the context and diverse meanings of the arXiv dataset contribute to higher quality question and answering pairs. Therefore, it better promotes multitask tuning.

COT answer vs. Direct answer for numerical question answering. In Fig.6, we compare using COT answer with direct answer in the same training pipeline for the chart numerical question answering task. Using COT answers instead of direct answers increases the accuracy from 51.9% to 72.1%, with improvements across all chart types, especially in dot-line and line charts, where accuracy has increased by 22% and 26.6% respectively. This improvement indicates the effectiveness of COT answers in elevating the overall accuracy and performance across various chart types, which reflects that using COT answers teaches the model the reasoning steps and offloads the calculations to the backend system, thus boosting the model's mathematical computation ability.

Table 10: A comparison of the results of ChartAssistant without arXiv dataset on five tasks with base type charts, which indicates that the arXiv dataset significantly improve the performance of the alignment pre-training and mulittask instruction tuning.

	ChartQA		Chart-to-Text		Chart-to-Table			
Model	aug.	human	Pew	Statista	ChartQA	OpenCQA	MathQA	ReferQA
stage1 w/o arXiv	89.9	43.7	13.8	39.1	91.1	14.5	64.1	61.1
stage2 w/o arXiv	89.7	42.6	12.6	37.5	91.3	13.2	56.7	56.4
Ours-D	**91.3**	**45.3**	**14.0**	**40.2**	**92.0**	**14.9**	**72.1**	**64.2**

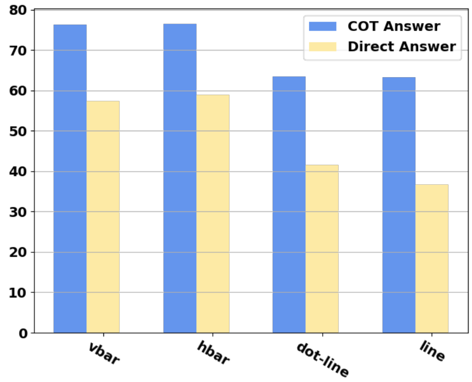


Figure 6: A comparison of the results of using COT answer and direct answer on numerical question answering task, which indicates that using COT answer significantly enhances the model's capability in handling chart numerical question answering tasks with all types.

Compared with Unichart after task-specific fine-tuning(except for Chart-to-Text). We employ the same training strategy and train with the identical model to highlight the effectiveness gains from our data. Following Unichart's lead in multi-task instruction tuning, as table 11 shows, we fine-tune the model on various test datasets (apart from Chart-to-Text, it utilizes fine-tuning during testing), resulting in improvements across different tasks surpassing those of Unichart. It is noteworthy that both Unichart and ChartAst-D are trained using Donut, emphasizing the superiority of ChartSFT.

The impact of each multitask instruction tuning component. We evaluated the impact of each segment in our multitask instruction tuning by excluding one task at a time during training and not-

Table 11: Compared with Unichart after task-specific fine-tuning.

Model	ChartQA aug.	ChartQA human	Chart-to-Text Pew	Chart-to-Text Statista	Chart-to-Table ChartQA	OpenCQA
Ours-D w/o align	89.0	42.1	13.7	38.3	89.5	14.3
Unichart	87.8	43.9	12.5	38.1	91.1	14.8
Ours-D w/o align(ft)	**89.6**	**44.2**	**13.7**	**38.3**	**91.4**	**14.9**

Table 12: ChartAssistant multitask instruction tuning ablations on ChartQA.

Model	ChartQA aug.	ChartQA human	ChartQA avg.
ChartAst-D	**91.3**	**45.3**	**68.3**
No Chart Summarization	90.0	43.5	66.7
No Open-ended Question Answering	89.5	41.1	65.3
No Numerical Question Answering	88.6	38.6	63.6
No Chart-to-Table Translation	88.8	41.0	64.9
No Referring Question Answering	89.2	41.2	65.2

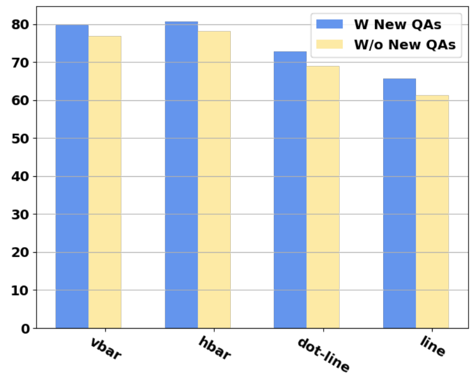


Figure 7: A comparison of the results of training with new question-answer pairs or not, which indicates that incorporating equivalent questions into the training process can enhance the model's robustness towards math questions.

ing effects on ChartQA performance. As table 12 shows, any omission led to a performance drop. In particular, chart summarization's contribution is smallest, possibly because ChartQA centers on data extraction and numerical question answering and not overall chart understanding. Furthermore, a significant performance decline when the numerical question answering task is excluded underlines its critical importance for the model.

Key components of ChartSFT analysis. For reasoning tasks involving specific numerical values, such as ChartQA, as shown in table 12, the math question-answering task benefits greatly from this, especially, as illustrated in fig .6, training in COT-format can significantly enhance the accuracy of mathematical computation problems. For tasks involving the output of long texts, such as openCQA, as demonstrated by table 4 and table 10, we find that incorporating a question-answering dataset composed of arXiv data can to some extent improve the performance of these tasks. We believe this is due to the broad scope, diversity, and specificity of the arXiv data. Moreover, compared to SciGraphQA (Li and Tajbakhsh, 2023), the arXiv data we provide has precise numerical values, results in higher quality question generation. Lastly, thanks to the robust language capabilities of GPT-3.5, it is capable of generating high-quality, comprehensive question-answering datasets.

The impact of generating equivalent math questions. Considering that generating questions purely through templates can be rather rigid in the math question answering task, we attempt to provide both the template questions and table information to ChatGPT simultaneously, asking it to generate more significant equivalent questions based on the meaning of the tables. In particular, "What is the difference between the highest and the lowest Amount of Least developed countries ?" can be converted to "What is the range of the Amount for Least developed countries ?". We divide these new question-answer pairs into training and test sets, then compare the performance on the test set when training with and without this additional data.

As fig. 7 demonstrates, we find that including the newly generated equivalent questions in the training can enhance the performance of all types compared to the original approach. In detail, the overall accuracy changes from 71.8% to 76.2%.

D Some demos from Out of Distribution

To demonstrate the model's generalization capability, we randomly take screenshots of several charts, as shown in Fig .9 and Fig .10 . We find that the model possesses generalization ability on out-of-distribution samples. Additionally, as shown in fig. 8, we visualize some demos comparing the performance of zero-shot scenarios with baseline methods. We observe that in summarization tasks, UniChart and MatCha tend to produce repetitions or hallucinations, whereas ChartLlama and ChartAssistant exhibit relatively stronger capabilities in handling summarization tasks. However, ChartLlama commits some factual errors; in question answering, thanks to the incorporation of COT-format QA training data, ChartAssistant effectively addresses QA tasks requiring mathematical reasoning. Lastly, in chart-to-table translation, UniChart and MatCha accurately model the table structure.

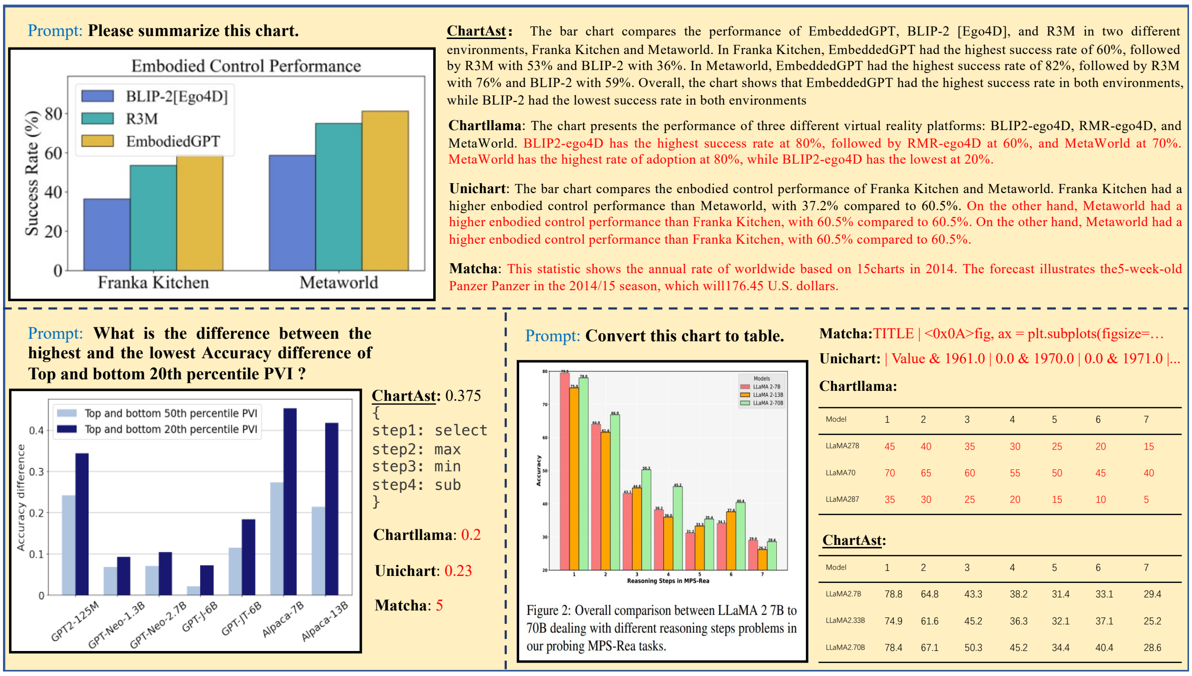


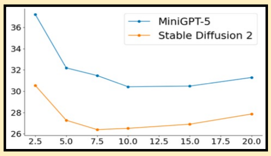
Figure 8: ChartAst-S demonstrates outstanding generalization ability in chart-to-table translation, summarization, and question-answering tasks.

Although ChartLlama can model the table structure accurately, the values are completely incorrect. Only ChartAssistant successfully constructs the table of the chart accurately.

Prompt: Please summarize the chart.

ChartAst: Analyzing the stock data from February 2023 to January 2024, we observe a bullish trend with some volatility. The stock opened at $4,265.18 on February 16, 2023, and showed a general upward trajectory with minor fluctuations. Initially, the stock experienced a slight dip, reaching a low of $4,187.63 on March 14, 2023. However, it recovered and entered a period of gradual growth, eventually reaching a high of $4,839.67 on December 22, 2023. The stock closed the year with a slight pullback, closing at $4,833.23 on December 29, 2023. In the new year, the stock continued its upward trajectory, reaching a peak closing price of $4,879.38 on January 5, 2024. Throughout the observed period, the stock showed resilience, with several dips and recoveries, indicating a robust market sentiment and potential investor confidence.

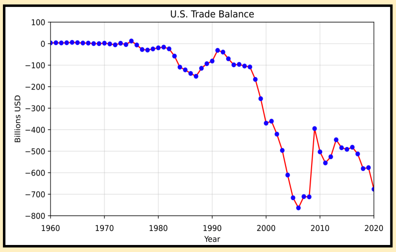
Prompt: Convert this chart to a table.



ChartAst:

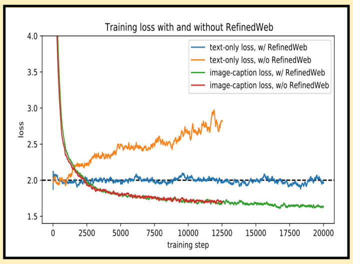
Entity	2.5	5.0	7.5	10.0	12.5	15.0	17.5	20.0
MiniGPT-5	37.0	32.1	31.2	30.3	30.3	30.3	30.6	31.2
Stable Diffusion 2	30.4	27.3	26.4	26.6	26.8	27.0	27.5	27.9

Prompt: Consider USD change over time and what conclusions can you draw from this.



ChartAst: The USD change over time shows a fluctuating trend, with a overall decrease from 1960 to 2020. The largest decrease occurred between 2000 and 2005, with a decrease of over 300 billion USD. This suggests that the U.S. has been running a trade deficit for most of the time period.

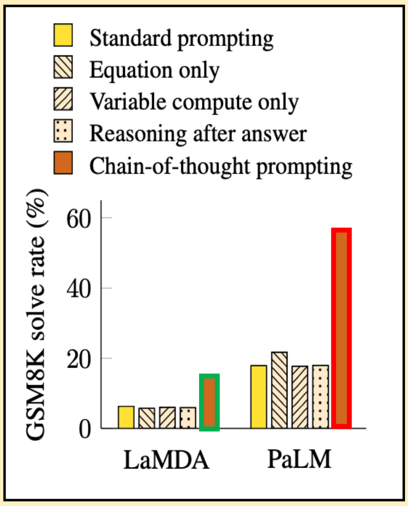
Prompt: Please answer my question based on the chart: What can you take away from considering whether to join RefinedWeb?



ChartAst: Considering whether to join RefinedWeb is an important decision that should be made carefully. The graph shows that RefinedWeb can help to improve the training process of the model, but it is important to weigh the benefits of RefinedWeb against the costs of implementing it.

Figure 9: ChartAst-S demonstrates outstanding generalization ability in chart-to-table translation, summarization, and question-answering tasks.

Prompt: Please use commandline to solve the math question about the referring box: What is the difference between theGSM8K solve rate (%) of green box and red box?

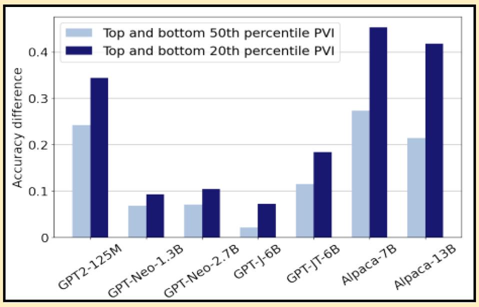


ChartAst:

```
{
 'step1':
 {
  'func1': 'select',
  'arg1': [
  'chain-of-thought
prompting',
  'chain-of-thought
prompting',
  'LaMDA', 'PaLM'
  ],
  'output1':
['16.500','57.600']
 },

 'step2':
 {
  'func2':
  'np.subtract',
  'arg2': 'output1',
  'output2': 'diff'
 }
}
```

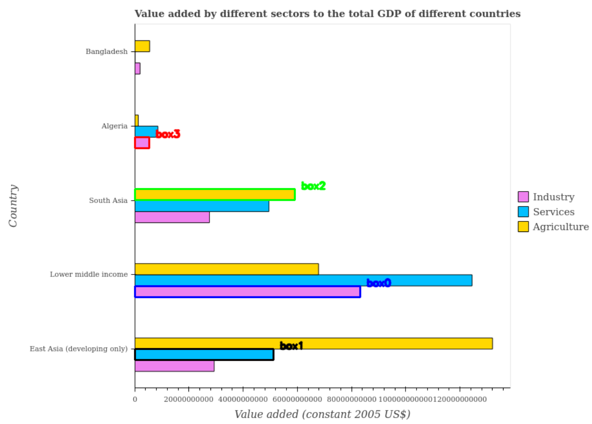
Prompt: Please use commandline to solve the math question: What is the difference between the highest and the lowest accuracy difference of Top and bottom 20th percentile PVI ?



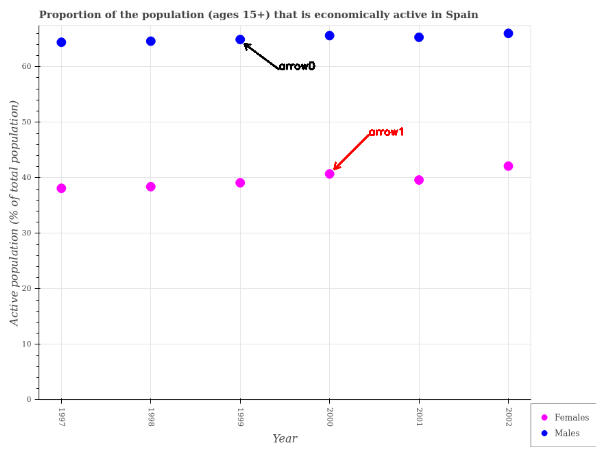
ChartAst:

```
{'step1': {'func1': 'select', 'arg1': 'top and
bottom 20th percentile PVI', 'output1': ['0.345',
'0.093', '0.104', '0.072', '0.191', '0.447',
'0.418']},
 'step2': {'func2': 'np.max', 'arg2': 'output1',
'output2': 'maximum'},
 'step3': {'func3': 'np.min', 'arg3': 'output1',
'output3': 'minimum'},
 'step4': {'func4': 'np.subtract', 'arg4':
['maximum', 'minimum'], 'output4': 'diff'}}
```

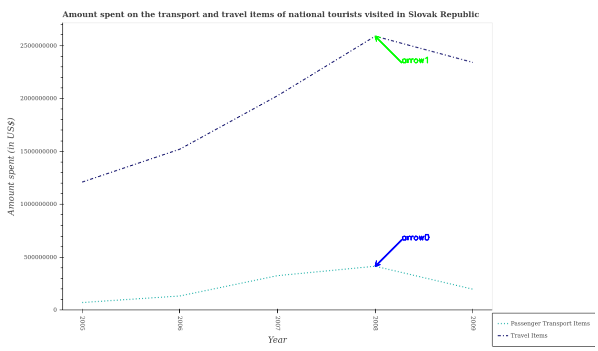
Figure 10: ChartAst-S demonstrates outstanding generalization ability in mathematical and referring question-answering tasks.



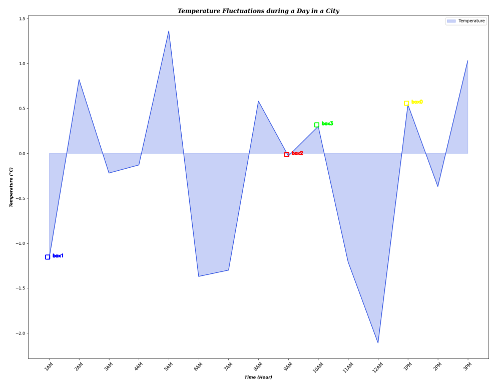
(a) bar chart with referring boxes



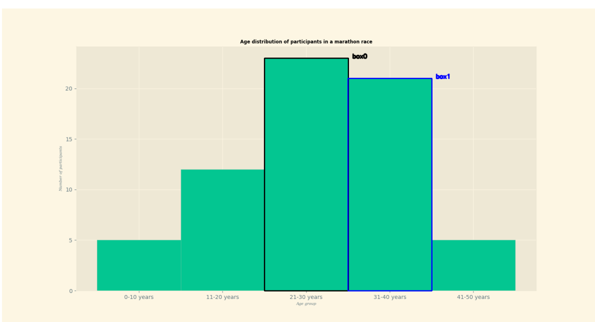
(b) dot-line chart with referring arrows



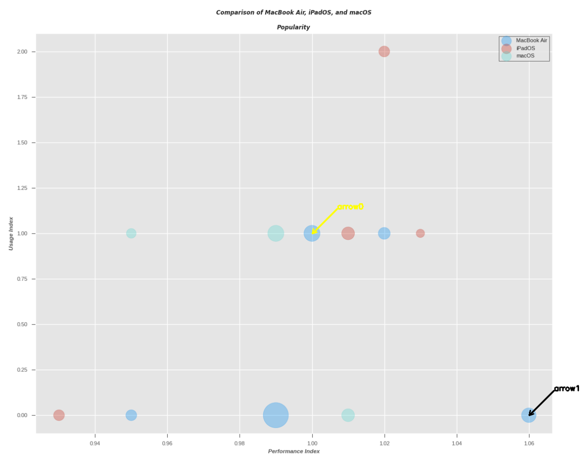
(c) line chart with referring arrows



(d) area chart with referring boxes



(e) histogram chart with referring boxes



(f) bubble chart with referring arrows

Figure 11: Some examples of different types of charts with referring markers.

NO.	Template	COT Steps	Func. Num.
	Templates in PlotQA		
1	What is the sum of <Y label> ?	2	2
2	What is the difference between the <Y label> in <ithx tick> and <jthx tick> ?	2	2
3	What is the average <Y label> per <singular form of X label> ?	2	2
4	What is the median <Y label> ?	2	2
5	What is the total <Y label> of/in <legend label> in the graph?	2	2
6	What is the difference between the <Y label> of/in <legend label> in <ithx tick> and that in <jthx tick> ?	2	2
7	What is the difference between the <Y label> of/in <legend label1> in <ithx tick> and the <Y label> of/in <legend label2> in <jthx tick> ?	2	2
8	What is the average <Y label> of/in <legend label> per <singular form of X label> ?	2	2
9	In the year <ithx tick> , what is the difference between the <Y label> of/in <legend label1> and<Y label> of/in <legend label2> ?	2	2
10	What is the difference between the <Y label> of/in <legend label1> and <Y label> of/in <legend label2> in <ithx tick> ?	2	2
11	In how many <plural form of X label> , is the<Y label> greater than <N> units ?	3	3
12	What is the ratio of the <Y label> in <ithx tick> to that in <jthx tick> ?	2	2
13	Is the <Y label> in <ithx tick> less than that in <jthx tick> ?	2	2
14	In how many <plural form of X label> , is the <Y label> of/in <legend label> greater than<N> <units> ?	3	3
15	What is the ratio of the <Y label> of/in <legend label> in <ithx tick> to that in <jthx tick> ?	2	2
16	Is the <Y label> of/in <legend label> in <ithx tick> less than that in <jthx tick> ?	2	2
17	Is the difference between the <Y label> in <ithx tick> and <jthx tick> greater than the difference between any two <plural form of X label> ?	8	6
18	What is the difference between the highest and the second highest <Y label> ?	6	5
19	Is the sum of the <Y label> of/in <legend label1> in <ithx tick> and <jthx tick> greater than the maximum<Y label> of/in <legend label2> across all <plural form of X label> ?	5	4
20	Is it the case that in every <singular form of X label> , the sum of the <Y label> of/in <legend label1> and <legend label2> is greater than the sum of <Y label> of <legend label3> and <Y label> of <legend label4> ?	8	4
21	Is the sum of the <Y label> in <ithx tick> and <jthx tick> greater than the maximum <Y label> across all <plural form of X label> ?	5	4
22	What is the difference between the highest and the lowest <Y label> ?	4	4
23	In how many <plural form of X label> , is the <Y label> greater than the average <Y label> taken over all <plural form of X label> ?	4	4
24	Is the difference between the <Y label> of/in <legend label1> in <ithx tick> and <jthx tick> greater than the difference between the <Y label> of/in <legend label2> in <ithx tick> and <jthx tick> ?	5	3
25	What is the difference between the highest and the second highest <Y label> of/in <legend label> ?	6	5
26	What is the difference between the highest and the lowest <Y label> of/in <legend label> ?	4	4

Figure 12: General Numerical QA Templates in ChartBench. Containing 40 template questions from PlotQA and 61 template questions that we designed additionally.

NO.	Template	COT Steps	Func. Num.
27	In how many <plural form of X label> , is the <Y label> of/in <legend label> greater than the average <Y label> of/in <legend label> taken over all <plural form of X label> ?	4	4
28	Is it the case that in every <singular form of X label> , the sum of the <Y label> of/in <legend label1> and <legend label2> is greater than the <Y label> of/in <legend label3> ?	6	4
29	Is the <Y label> of/in <legend label1> strictly greater than the <Y label> of/in <legend label2> over the <plural form of X label> ?	4	3
30	Is the <Y label> of/in <legend label1> strictly less than the <Y label> of/in <legend label2> over the <plural form of X label> ?	4	3
31	Does the <Y label> of/in <legend label> monotonically increase over the <plural form of X label> ?	4	4
32	What is the difference between two consecutive major ticks on the Y-axis ?	4	3
33	Across all <plural form of X label> , what is the maximum <Y label> ?	2	2
34	Across all <plural form of X label> , what is the minimum <Y label> ?	2	2
35	In which <X label> was the <Y label> maximum ?	4	3
36	In which <X label> was the <Y label> minimum ?	4	3
37	Across all <plural form of X label> , what is the maximum <Y label> of/in <legend label> ?	2	2
38	Across all <plural form of X label> , what is the minimum <Y label> of/in <legend label> ?	2	2
39	In which <singular form of X label> was the <Y label> of/in <legend label> maximum ?	4	3
40	In which <singular form of X label> was the <Y label> of/in <legend label> minimum ?	4	3
	Extended Templates		
41	Across all <plural form of X label> , what is the covariance between the <Y label> of/in <legend label1> and <Y label> of/in <legend label2> ?	3	2
42	Across all <plural form of X label> , what is the correlation coefficient between the <Y label> of/in <legend label1> and <Y label> of/in <legend label2> ?	3	2
43	What is the percentage change in the <Y label> of/in <legend label> from <ithx tick> to <jthx tick> ?	4	4
44	Across all <plural form of X label> , what is the percentage of the <Y label> of/in <legend label> which below <N> <units> ?	5	5
45	What is the sum of the <Y label> of/in <legend label> with <plural form of X label> in the range of <ithx tick> to <jthx tick> ?	2	2
46	What is the average change in <Y label> of/in <legend label> between consecutive <plural form of X label> ?	3	3
47	What is the median <Y label> of/in <legend label> in the graph?	2	2
48	What is the ratio between the highest and the lowest <Y label> of/in <legend label> ?	4	4
49	What is the ratio between the highest and the second lowest <Y label> of/in <legend label> ?	5	4
50	What is the ratio of the difference between the maximum and minimum <Y label> of/in <legend label> to the average <Y label> of/in <legend label> ?	6	6
51	For <legend label> , is the highest <Y label> greater than three times the lowest <Y label> ?	5	5
52	For <legend label> , is the difference between maximum and minimum of <Y label> greater than the sum of the mean and median <Y label> ?	8	8

Figure 13: – continued from previous page.

NO.	Template	COT Steps	Func. Num.
53	What is the standard deviation of <Y label> ?	2	2
54	Is the sum of <Y label> of/in <legend label> in <plural form of X label> strictly greater than <N> <units> ?	3	3
55	What is the difference between the mean <Y label> of/in <legend label> and the median <Y label> of/in <legend label> across all <plural form of X label> ?	4	4
56	Is the maximum <Y label> of/in <legend label> greater than four times the minimum <Y label> of/in <legend label> ?	5	5
57	For <legend label1> and <legend label2> , which one was the median <Y label> maximum ?	6	4
58	For <legend label1> , <legend label2> and <legend label3> , which one was the average <Y label> maximum across all <plural form of X label> ?	8	4
59	For <legend label1> , <legend label2> and <legend label3> , which one was the sum <Y label> minimum across all <plural form of X label> ?	8	4
60	Among <legend label1> and <legend label2> , which one has the smallest difference between the maximum and minimum <Y label> across all <plural form of X label> ?	10	6
61	Among <legend label1> and <legend label2> , which one has the biggest difference between the maximum and minimum <Y label> across all <plural form of X label> ?	10	6
62	Among <legend label1> , <legend label2> , which one has the smallest absolute difference between the median and mean <Y label> ?	12	7
63	Across all <plural form of X label> , are the <Y label> values of <legend label1> and <legend label2> positively correlated?	4	3
64	What is the standard deviation of <Y label> of/in <legend label> ?	2	2
65	Across all <plural form of X label> , are the <Y label> values of <legend label1> and <legend label2> negatively correlated?	4	3
66	Among <legend label1> , <legend label2> , and <legend label3> , which one has the smallest standard deviation of <Y label> across all <plural form of X label> ?	8	4
67	Among <legend label1> , <legend label2> , and <legend label3> , which one has the biggest variance of <Y label> across all <plural form of X label> ?	8	4
68	What is the difference between the mean <Y label> for <legend label1> in the range of <ithx tick> to <jthx tick> and that for <legend label2> ?	5	3
69	Is the correlation between <Y label> of/in <legend label1> and <legend label2> stronger than the correlation between <Y label> of/in <legend label1> and <legend label3> across all <plural form of X label> ?	8	4
70	In How many <plural form of X label> ,the <Y label> of/in <legend label1> greater than twice the mean of the <Y label> of/in <legend label2> ?	6	5
71	What is the difference between the uppermost/rightmost and bottommost/leftmost <Y label> of/in <legend label> in the graph?	4	3
72	What is the difference between the uppermost/rightmost and second uppermost/rightmost <Y label> of/in <legend label> in the graph?	4	3
73	What is the ratio between the bottommost/leftmost and second bottommost/leftmost <Y label> of/in <legend label> in the graph?	4	3
74	What is the sum between the bottommost/leftmost and second bottommost/leftmost <Y label> of/in <legend label> in the graph?	4	3
75	What is the ratio of the sum of <Y label> of/in <legend label> in <ithx tick> and <jthx tick> to the difference between the <Y label> of/in <legend label> in <ithx tick> and <jthx tick> ?	4	4
76	What is the product of the highest and the lowest <Y label> of/in <legend label> ?	4	4

Figure 14: – continued from previous page.

NO.	Template	COT Steps	Func. Num.
77	What is the product of the <Y label> of/in <legend label> in <ithx tick> and <jthx tick> ?	2	2
78	In How many <plural form of X label> ,the <Y label> of/in <legend label1> strictly less than twice the mean of the <Y label> of/in <legend label2> ?	6	5
79	Among <legend label1> , <legend label2> , and <legend label3> , which one has the biggest mean <Y label> across all <plural form of X label> ?	8	4
80	Among <legend label1> , <legend label2> , and <legend label3> , which one has the biggest median <Y label> across all <plural form of X label> ?	8	4
81	Among <legend label1> , <legend label2> , and <legend label3> , which one has the smallest total <Y label> across all <plural form of X label> ?	8	4
82	How much is three times the average <Y label> of/in <legend label> per <singular form of X label> ?	3	3
83	How much is twice the sum <Y label> of/in <legend label> across all <plural form of X label> ?	3	3
84	What is the average of the maximum and minimum <Y label> in <legend label> across all <plural form of X label> ?	4	4
85	Is the maximum <Y label> of/in <legend label> less than twice the median <Y label> of/in <legend label> across all <plural form of X label> ?	5	5
86	What is the difference between the average <Y label> per <singular form of X label> in <legend label1> and the average <Y label> per <singular form of X label> in <legend label2> ?	5	3
87	Between the <Y label> of/in <legend label1> and <legend label2> , which one has the higher average change?	8	5
88	What is the variance of <Y label> ?	2	2
89	What is the variance of <Y label> of/in <legend label> ?	2	2
90	What is the product of the mean and the median <Y label> of/in <legend label> ?	4	4
91	Is the median <Y label> of/in <legend label> less than the mean <Y label> of/in <legend label> across all <plural form of X label> ?	4	4
92	What is the ratio of the difference between the maximum and minimum <Y label> of/in <legend label> to the standard deviation of <Y label> of/in <legend label> ?	6	6
93	What is the difference between the uppermost/rightmost <Y label> of/in <legend label> and highest <Y label> of/in <legend label> in the graph?	4	4
94	What is the difference between the bottommost/leftmost <Y label> of/in <legend label> and lowest <Y label> of/in <legend label> in the graph?	4	4
95	What is the ratio of the mean <Y label> of/in <legend label> to the standard deviation of <Y label> of/in <legend label> ?	4	4
96	Is the difference between the maximum and minimum <Y label> of/in <legend label> within the range of <N1> to <N2> ?	7	7
97	Is the sum of the <Y label> of/in <legend label> in <ithx tick> and <jthx tick> greater than twice the difference between the <Y label> of/in <legend label> in <ithx tick> and <jthx tick> ?	5	5
98	Across all <plural form of X label> , is the median <Y label> of/in <legend label> within the range of <N1> to <N2> ?	5	5
99	Is the difference between the <Y label> of/in <legend label> in <ithx tick> and <jthx tick> greater than the sum of the <Y label> of/in <legend label> in <kthx tick> and <lthx tick> ?	5	4
100	What is the difference between the maximum <Y label> of/in <legend label1> and the maximum <Y label> of/in <legend label2> ?	5	3
101	What is the average <Y label> of/in <legend label> with <plural form of X label> in the range of <ithx tick> to <jthx tick> ?	2	2

Figure 15: – continued from previous page.

NO.	Template	COT Steps	Func. Num.
	Box-plot		
1	What is the ratio of the <Y label> in <legend label1> to that in <legend label2> ?	2	2
2	What is the product of the highest and the lowest <Y label> ?	4	4
3	What is the product of the mean and the median <Y label> ?	4	4
4	What is the ratio of the difference between the maximum and minimum <Y label> to the standard deviation of <Y label> ?	6	6
5	What is the difference between the <Y label> in <legend label1> and <legend label2> ?	2	2
6	Is the <Y label> in <legend label1> less than that in <legend label2> ?	2	2
7	What is the ratio between the highest and the lowest <Y label> ?	4	4
8	What is the ratio between the highest and the second lowest <Y label> ?	5	4
9	What is the ratio of the difference between the maximum and minimum <Y label> to the average <Y label> ?	6	6
10	Is the highest <Y label> greater than <N> times the lowest <Y label> ?	5	5
11	Is the difference between maximum and minimum of <Y label> greater than the sum of the mean and median <Y label> ?	8	8
	Bubble Chart		
12	What is the average <X label> of/in <legend label> ?	2	2
13	What is the total <X label> of/in <legend label> in the graph?	2	2
14	What is the difference between the highest and the lowest <X label> of/in <legend label> ?	4	4
15	What is the minimum <X label> of/in <legend label> ?	2	2
16	What is the covariance between the <Y label> and <Z label> of/in <legend label> ?	3	2
17	What is the correlation coefficient between the <X label> and <Z label> of/in <legend label> ?	3	2
18	What is the ratio between the highest and the second lowest <X label> of/in <legend label> ?	5	4
19	For <legend label> , is the difference between maximum and minimum of <X label> greater than the sum of the mean and median <X label> ?	8	8
20	Is the maximum <X label> of/in <legend label> greater than four times the minimum <X label> of/in <legend label> ?	5	5
21	Among <legend label1> and <legend label2> , which one has the biggest difference between the maximum and minimum <X label> ?	10	6
22	Are the <Y label> and <Z label> of/in <legend label> positively correlated?	4	3
23	Are the <X label> and <Z label> of/in <legend label> negatively correlated?	4	3
24	What is the average of the maximum and minimum <X label> in <legend label> ?	4	4
25	What is the variance of <X label> of/in <legend label> ?	2	2
26	What is the product of the mean and the median <X label> of/in <legend label> ?	4	4
27	What is the ratio of the mean <X label> of/in <legend label> to the standard deviation of <X label> of/in <legend label> ?	4	4
28	Is the difference between the maximum and minimum <X label> of/in <legend label> within the range of <N1> to <N2> ?	7	7
29	What is the difference between the maximum <X label> of/in <legend label1> and the maximum <X label> of/in <legend label2> ?	5	3
	Histogram		
30	What is the proportion of <Y label> of/in <ithx tick> if the <Y label> of/in <ithx tick> become <N> times the original?	7	6

Figure 16: Numerical QA Templates for several Types of charts in ChartBench.

NO.	Template	COT Steps	Func. Num.
31	What is the <Y label> of/in <ithx tick> if the total <Y label> become <N> times the original?	2	2
32	If total <Y label> is <N> , how many <Y label> would be in <ithx tick> ?	5	4
33	If <ithx tick> is removed, what would be the new percentage of <Y label> of/in <jthx tick> ?	6	4
34	What is the proportion of <Y label> of/in <ithx tick> ?	4	3
35	What is the proportion of <Y label> with <ithx tick> and above ?	5	3
36	What is the proportion of <Y label> with <ithx tick> and below ?	5	3
37	What is the proportion of <Y label> in the range of <ithx tick> to <jthx tick> ?	5	3
	Pie Chart		
38	What is the proportion of <Y label> of/in <ithx tick> if the <Y label> of/in <ithx tick> become <N> times the original?	7	6
39	What is <Y label> of/in <ithx tick> if the total <Y label> become <N> times the original?	2	2
40	If total <Y label> is <N> , how many <Y label> would be in <ithx tick> ?	5	4
41	If <ithx tick> is removed, what would be the new percentage of <Y label> of/in <jthx tick> ?	6	4
42	What is the proportion of <Y label> of/in <ithx tick> ?	4	3
43	What is the total proportion of <Y label> of/in <ithx tick> and <jthx tick> ?	6	4

Figure 17: – continued from previous page.

NO.	Template	COT Steps	Func. Num.
	General		
1	What does the <box0> represent?	-	-
2	What does the <color0> box represent?	-	-
3	What does the <arrow0> represent?	-	-
4	What does the <color0> arrow represent?	-	-
5	What is the label of <box0> ?	-	-
6	What is the label of <color0> box?	-	-
7	What is the label of <arrow0> ?	-	-
8	What is the label of <color0> arrow?	-	-
9	Across all <plural form of X label> , what is the maximum <Y label> of the legend represented by the <arrow0> ?	2	2
10	Across all <plural form of X label> , what is the maximum <Y label> of the legend represented by the <color0> arrow?	2	2
11	Across all <plural form of X label> , what is the maximum <Y label> of the legend represented by the <box0> ?	2	2
12	Across all <plural form of X label> , what is the maximum <Y label> of of the legend represented by the <color0> box?	2	2
13	Across all <plural form of X label> , what is the maximum <Y label> of the <color0> arrows?	2	2
14	Across all <plural form of X label> , what is the maximum <Y label> of the <color0> boxes?	2	2
15	Across all <plural form of X label> , what is the minimum <Y label> of the legend represented by the <arrow0> ?	2	2

Figure 18: Referring QA Templates in ChartBench.

NO.	Template	COT Steps	Func. Num.
16	Across all <plural form of X label> , what is the minimum <Y label> of the legend represented by the <color0> arrow?	2	2
17	Across all <plural form of X label> , what is the minimum <Y label> of the legend represented by the <box0> ?	2	2
18	Across all <plural form of X label> , what is the minimum <Y label> of the legend represented by the <color0> box?	2	2
19	Across all <plural form of X label> , what is the minimum <Y label> of the <color0> arrows?	2	2
20	Across all <plural form of X label> , what is the minimum <Y label> of the <color0> boxes?	2	2
21	What is the average <Y label> of the legend represented by the <arrow0> per <X label> ?	2	2
22	What is the average <Y label> of the legend represented by the <color0> arrow per <X label> ?	2	2
23	What is the average <Y label> of the legend represented by the <box0> per <X label> ?	2	2
24	What is the average <Y label> of the legend represented by the <color0> box per <X label> ?	2	2
25	What is the average <Y label> of the <color0> arrows per <X label> ?	2	2
26	What is the average <Y label> of the <color0> boxes per <X label> ?	2	2
27	What is the median <Y label> of the legend represented by the <arrow0> per <X label> ?	2	2
28	What is the median <Y label> of the legend represented by the <color0> arrow per <X label> ?	2	2
29	What is the median <Y label> of the legend represented by the <box0> per <X label> ?	2	2
30	What is the median <Y label> of the legend represented by the <color0> box per <X label> ?	2	2
31	What is the median <Y label> of the <color0> arrows per <X label> ?	2	2
32	What is the median <Y label> of the <color0> boxes per <X label> ?	2	2
33	What is the total <X label> of the legend represented by the <arrow0> in the graph?	2	2
34	What is the total <X label> of the legend represented by the <color0> arrow in the graph?	2	2
35	What is the total <X label> of the legend represented by the <box0> in the graph?	2	2
36	What is the total <X label> of the legend represented by the <color0> box in the graph?	2	2
37	What is the total <Y label> of the <color0> arrows?	2	2
38	What is the total <Y label> of the <color0> boxes?	2	2
39	In how many <plural form of X label> , is the <Y label> of the legend represented by the <arrow0> greater than the average <Y label> of it taken over all <plural form of X label> ?	4	4
40	In how many <plural form of X label> , is the <Y label> of the legend represented by the <color0> arrow greater than the average <Y label> of it taken over all <plural form of X label> ?	4	4
41	In how many <plural form of X label> , is the <Y label> of the legend represented by the <box0> greater than the average <Y label> of it taken over all <plural form of X label> ?	4	4
42	In how many <plural form of X label> , is the <Y label> of the legend represented by the <color0> box greater than the average <Y label> of it taken over all <plural form of X label> ?	4	4

Figure 19: – continued from previous page.

NO.	Template	COT Steps	Func. Num.
43	In how many <plural form of X label> , is the <Y label> of the legend represented by the <arrow0> less than the average <Y label> of it taken over all <plural form of X label> ?	4	4
44	In how many <plural form of X label> , is the <Y label> of the legend represented by the <color0> arrow less than the average <Y label> of it taken over all <plural form of X label> ?	4	4
45	In how many <plural form of X label> , is the <Y label> of the legend represented by the <box0> less than the average <Y label> of it taken over all <plural form of X label> ?	4	4
46	In how many <plural form of X label> , is the <Y label> of the legend represented by the <color0> box less than the average <Y label> of it taken over all <plural form of X label> ?	4	4
47	Is the <Y label> of the legend represented by the <arrow0> strictly greater than the <Y label> of the legend represented by the <arrow1> over the <plural form of X label> ?	4	3
48	Is the <Y label> of the legend represented by the <color0> arrow strictly greater than the <Y label> of the legend represented by the <color1> arrow over the <plural form of X label> ?	4	3
49	Is the <Y label> of the legend represented by the <box0> strictly greater than the <Y label> of the legend represented by the <box1> over the <plural form of X label> ?	4	3
50	Is the <Y label> of the legend represented by the <color0> box strictly greater than the <Y label> of the legend represented by the <color1> box over the <plural form of X label> ?	4	3
51	Is the <Y label> of the legend represented by the <arrow0> strictly less than the <Y label> of the legend represented by the <arrow1> over the <plural form of X label> ?	4	3
52	Is the <Y label> of the legend represented by the <color0> arrow strictly less than the <Y label> of the legend represented by the <color1> arrow over the <plural form of X label> ?	4	3
53	Is the <Y label> of the legend represented by the <box0> strictly less than the <Y label> of the legend represented by the <box1> over the <plural form of X label> ?	4	3
54	Is the <Y label> of the legend represented by the <color0> box strictly less than the <Y label> of the legend represented by the <color1> box over the <plural form of X label> ?	4	3
55	What is the difference between the <Y label> of <box0> and <box1> ?	2	2
56	What is the difference between the <Y label> of <arrow0> and <arrow1> ?	2	2
57	What is the difference between the <Y label> of <color0> box and <color1> box?	2	2
58	What is the difference between the <Y label> of <color0> arrow and <color1> arrow?	2	2
59	What is the ratio of the <Y label> of <box0> to that of <box1> ?	2	2
60	What is the ratio of the <Y label> of <color0> box to that of <color1> box?	2	2
61	What is the ratio of the <Y label> of <arrow0> to that of <arrow1> ?	2	2
62	What is the ratio of the <Y label> of <color0> arrow to that of <color1> arrow?	2	2
63	Is the <Y label> of <box0> less than that of <box1> ?	2	2
64	Is the <Y label> of <color0> box less than that of <color1> box?	2	2
65	Is the <Y label> of <arrow0> less than that of <arrow1> ?	2	2
66	Is the <Y label> of <color0> arrow less than that of <color1> arrow?	2	2
67	Is the difference between the <Y label> of <box0> and <box1> greater than the difference of the legend represented by the <box0> between any two <plural form of X label> ?	8	6

Figure 20: – continued from previous page.

NO.	Template	COT Steps	Func. Num.
68	Is the difference between the <Y label> of <color0> box and <color1> box greater than the difference of the legend represented by the <color0> box between any two <plural form of X label> ?	8	6
69	Is the difference between the <Y label> of <arrow0> and <arrow1> greater than the difference of the legend represented by the <arrow0> between any two <plural form of X label> ?	8	6
70	Is the difference between the <Y label> of <color0> arrow and <color1> arrow greater than the difference of the legend represented by the <color0> arrow between any two <plural form of X label> ?	8	6
71	Is the sum of the <Y label> of <box0> and <box1> greater than the maximum <Y label> of the legend represented by the <box0> across all <plural form of X label> ?	5	4
72	Is the sum of the <Y label> of <color0> box and <color1> box greater than the maximum <Y label> of the legend represented by the <color0> across all <plural form of X label> ?	5	4
73	Is the sum of the <Y label> of <arrow0> and <arrow1> greater than the maximum <Y label> of the legend represented by the <arrow0> across all <plural form of X label> ?	5	4
74	Is the sum of the <Y label> of <color0> arrow and <color1> arrow greater than the maximum <Y label> of the legend represented by the <color0> across all <plural form of X label> ?	5	4
75	Is the difference between the <Y label> of <box0> and <box1> greater than the difference between the <Y label> of <box2> and <box3> ?	7	4
76	Is the difference between the <Y label> of <color0> box and <color1> box greater than the difference between the <Y label> of <color2> box and <color3> box?	7	4
77	Is the difference between the <Y label> of <arrow0> and <arrow1> greater than the difference between the <Y label> of <arrow2> and <arrow3> ?	7	4
78	Is the difference between the <Y label> of <color0> arrow and <color1> arrow greater than the difference between the <Y label> of <color2> arrow and <color3> arrow?	7	4
79	Is it the case that in every <X label> , the sum of the <Y label> of the legend represented by the <arrow0> and the legend represented by the <arrow1> is greater than the <Y label> of the legend represented by the <arrow2> ?	6	4
80	Is it the case that in every <X label> , the sum of the <Y label> of the legend represented by the <color0> arrow and the legend represented by the <color1> arrow is greater than the <Y label> of the legend represented by the <color2> arrow?	6	4
81	Is it the case that in every <X label> , the sum of the <Y label> of the legend represented by the <box0> and the legend represented by the <box1> is greater than the <Y label> of the legend represented by the <box2> ?	6	4
82	Is it the case that in every <X label> , the sum of the <Y label> of the legend represented by the <color0> box and the legend represented by the <color1> box is greater than the <Y label> of the legend represented by the <color2> box?	6	4
	Line and Area Chart		
83	Across all <plural form of X label> , what is the maximum <Y label> of this <arrow0> ?	2	2
84	Across all <plural form of X label> , what is the maximum <Y label> of this <color0> arrow?	2	2
85	Across all <plural form of X label> , what is the minimum <Y label> of this <arrow0> ?	2	2
86	Across all <plural form of X label> , what is the minimum <Y label> of this <color0> arrow?	2	2

Figure 21: – continued from previous page.

NO.	Template	COT Steps	Func. Num.
87	What is the average <Y label> of <arrow0> per <X label> ?	2	2
88	What is the average <Y label> of <color0> arrow per <X label> ?	2	2
89	What is the median <Y label> of <arrow0> per <X label> ?	2	2
90	What is the median <Y label> of <color0> arrow per <X label> ?	2	2
91	What is the total <Y label> of <arrow0> in the graph?	2	2
92	What is the total <Y label> of <color0> arrow in the graph?	2	2
93	In how many <plural form of X label> , is the <Y label> of <arrow0> greater than the average <Y label> of <arrow0> taken over all <plural form of X label> ?	4	4
94	In how many <plural form of X label> , is the <Y label> of <color0> arrow greater than the average <Y label> of <color0> arrow taken over all <plural form of X label> ?	4	4
95	In how many <plural form of X label> , is the <Y label> of <arrow0> less than the average <Y label> of <arrow0> taken over all <plural form of X label> ?	4	4
96	In how many <plural form of X label> , is the <Y label> of <color0> arrow less than the average <Y label> of <color0> arrow taken over all <plural form of X label> ?	4	4
97	Is the <Y label> of <arrow0> strictly greater than the <Y label> of <arrow1> over the <plural form of X label> ?	4	3
98	Is the <Y label> of <color0> arrow strictly greater than the <Y label> of <color1> arrow over the <plural form of X label> ?	4	3
99	Is the <Y label> of <arrow0> strictly less than the <Y label> of <arrow1> over the <plural form of X label> ?	4	3
100	Is the <Y label> of <color0> arrow strictly less than the <Y label> of <color1> arrow over the <plural form of X label> ?	4	3
101	What is the difference between the <Y label> of <box0> and <box1> ?	2	2
102	What is the difference between the <Y label> of <color0> box and <color1> box?	2	2
103	What is the ratio of the <Y label> of <box0> to that of <box1> ?	2	2
104	What is the ratio of the <Y label> of <color0> box to that of <color1> box?	2	2
105	Is the <Y label> of <box0> less than that of <box1> ?	2	2
106	Is the <Y label> of <color0> box less than that of <color1> box?	2	2
107	Is the difference between the <Y label> of <box0> and <box1> greater than the difference of the corresponding legend of <box0> between any two <plural form of X label> ?	8	6
108	Is the difference between the <Y label> of <color0> box and <color1> box greater than the difference of the corresponding legend of <color0> box between any two <plural form of X label> ?	8	6
109	Is the sum of the <Y label> of <box0> and <box1> greater than the maximum <Y label> of the corresponding legend of <box0> across all <plural form of X label> ?	5	4
110	Is the sum of the <Y label> of <color0> box and <color1> box greater than the maximum <Y label> of the corresponding legend of <color0> across all <plural form of X label> ?	5	4
111	Is the difference between the <Y label> of <box0> and <box1> greater than the difference between the <Y label> of <box2> and <box3> ?	7	4
112	Is the difference between the <Y label> of <color0> box and <color1> box greater than the difference between the <Y label> of <color2> box and <color3> box?	7	4
113	Is it the case that in every <X label> , the sum of the <Y label> of <arrow0> and <arrow1> is greater than the <Y label> of <arrow2> ?	6	4
114	Is it the case that in every <X label> , the sum of the <Y label> of <color0> arrow and <color1> arrow is greater than the <Y label> of <color2> arrow?	6	4

Figure 22: – continued from previous page.