ChartGemma: Visual Instruction-tuning for Chart Reasoning in the Wild

Ahmed Masry[♣*] **Megh Thakkar**[♥*] **Aayush Bajaj**[♥†] **Aaryaman Kartha**[♣†]
Enamul Hoque[♣] **Shafiq Joty**[♦♠]
[♣]York University, Canada [♥]Chandar Research Lab; MILA - Quebec AI Institute
[♦]Salesforce Research [♠]Nanyang Technological University, Singapore
{masry20, aarykary, enamulh}@yorku.ca
{megh.thakkar, aayush.bajaj}@mila.quebec, sjoty@salesforce.com

Abstract

Given the ubiquity of charts as a data analysis, visualization, and decision-making tool across industries and sciences, there has been a growing interest in developing pre-trained foundation models as well as general purpose instruction-tuned models for chart understanding and reasoning. However, existing methods suffer crucial drawbacks across two critical axes affecting the performance of chart representation models: they are trained on data generated from underlying data tables of the charts, ignoring the visual trends and patterns in chart images, *and* use weakly aligned vision-language backbone models for domain-specific training, limiting their generalizability when encountering charts in the wild. We address these important drawbacks and introduce ChartGemma, a novel chart understanding and reasoning model developed over PaliGemma. Rather than relying on underlying data tables, ChartGemma is trained on instruction-tuning data generated directly from chart images, thus capturing both high-level trends and low-level visual information from a diverse set of charts. Our simple approach achieves state-of-the-art results across 5 benchmarks spanning chart summarization, question answering, and fact-checking, and our elaborate qualitative studies on real-world charts show that ChartGemma generates more realistic and factually correct summaries compared to its contemporaries. We release the code, model checkpoints, dataset, and demos at https://github.com/vis-nlp/ChartGemma.[1]

1 Introduction

Language-augmented vision foundation models or vision-language models (VLMs) have proven to be effective in tackling numerous real-world multimodal tasks such as visual segmentation, captioning, question answering, and generation and editing (Li et al., 2023; Zhu et al., 2023). Though these models excel when used for general purpose applications in the wild, they often fail to tackle tasks that require specialized understanding and decoding of patterns and visualizations (Han et al., 2023). An important domain-specific usage of VLMs is for understanding and reasoning over charts, given their ubiquity as a data analysis, visualization, and decision-making tool across businesses, economies, and scientific fields (Hoque et al., 2022). This has naturally led to the development of more specialized foundation models pre-trained on massive amounts of structured and often chart-specific data (Liu et al., 2022; Masry et al., 2023). These models are, however, trained on a limited source of resources and focus on a specific set of tasks, constraining their real-world applicability (Masry et al., 2024).

Developing over the success of instruction-tuning enabling models to generalize to more tasks and applications (Ouyang et al., 2022), there have been attempts at 'instruction-tuning' VLMs to endow them the ability to understand charts in more realistic and fundamental settings (Meng et al., 2024). These approaches generally depend on two crucial factors impacting their effectiveness: (i) Instruction-tuning dataset – these methods either use the underlying data tables from existing web sources (Masry et al., 2024) or use synthetically generated data-tables (Han et al., 2023) from LLMs such as GPT-4 (OpenAI, 2023) to curate the instruction-tuning data, and (ii) Base model – the existing methods either use chart-specific pre-trained models like UniChart (Masry et al., 2023) or VLMs pre-trained with weak image-text alignment such as LLaVA (Li et al., 2023). However, in existing methods, both these factors have critical drawbacks impacting their ability to understand real-world complex charts.

Existing methods are restricted to charts that ei-

* †Equal contribution.

[1]Preprint: https://arxiv.org/abs/2407.04172

Proceedings of the 31st International Conference on Computational Linguistics: Industry Track, pages 625–643
January 19–24, 2025.

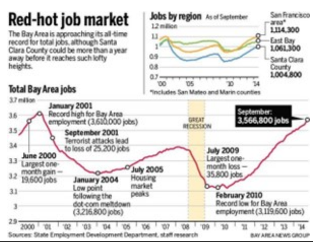
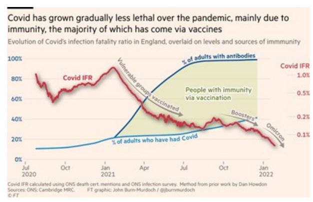
<table>
<tr><th>Chart Image</th><th>Summary Generated from the Data Table</th><th>Summary Generated from Chart Image</th></tr>
<tr><td>
</td><td>The provided data table shows the total number of jobs in the Bay Area, as well as the number of jobs in San Francisco, the East Bay, and Santa Clara County, from 2000 to 2026. Unfortunately, the table only provides data for the year 2000, with the remaining years missing data for all regions except the total Bay Area jobs. While we can see the total Bay Area job growth from 3.3 million in 2000 to a projected 4.0 million in 2026, we lack the specific data to assess job trends within San Francisco, East Bay, and Santa Clara County over the years. To draw meaningful conclusions about job market trends in these regions, it is necessary to have complete data for all years and all regions.</td><td>The chart shows the job market in the Bay Area from 2000 to 2012. The Bay Area is approaching its all-time record for total jobs, although Santa Clara County could be more than a year away before it reaches such lofty heights. The chart shows that the Bay Area job market experienced a significant drop during the Great Recession, but has since recovered and is now at an all-time high. The chart also shows the number of jobs in different regions of the Bay Area, including San Francisco, East Bay, and Santa Clara County. The chart is a good indicator of the overall health of the Bay Area economy.</td></tr>
<tr><td>
</td><td>The data table shows the trend of Covid-19 related metrics from July 2020 to January 2022. It reveals a decline in the Infection Fatality Rate (IFR) from 80% in July 2020 to 25% in January 2022. This coincides with a rise in the percentage of adults who have had Covid, increasing from 10% in July 2020 to 60% in January 2022. The percentage of adults with antibodies also shows a significant increase, going from 0% in July 2020 to 100% in January 2022. These trends suggest a possible correlation between increased exposure to Covid-19 and a lower IFR, likely due to herd immunity and improved treatment options.</td><td>This chart shows the evolution of Covid's infection fatality ratio in England, overlaid on levels and sources of immunity. The chart shows that the Covid infection fatality ratio has decreased significantly since the beginning of the pandemic. This is likely due to the increasing levels of immunity in the population, both from vaccination and from infection. The chart also shows that the Covid infection fatality ratio has been declining at a faster rate since the introduction of vaccines. It also shows that the most vulnerable groups have been vaccinated and that this has had a significant impact on reducing the number of deaths from Covid. The infection fatality ratio has also decreased with the introduction of boosters. This demonstrates the impact of the vaccine in helping to protect people from serious illness and death from Covid.</td></tr>
</table>

Table 1: Summaries generated from the same LLM, Gemini Flash 1.5, when using the data table and the chart image, highlighting the importance of understanding the visual attributes to generate more appropriate chart instructions.

ther have an underlying data table or require methods to extract them from the charts, often with low accuracy which are used for instruction-tuning data generation. These data tables are often incapable of capturing numerous nuanced details in the complex charts used in real-world applications (Table 1). Also, in many scenarios, we are concerned with representing or understanding general trends in the charts and not individual data points. On the model side, existing methods use backbones in which the vision encoder and LLM are weakly-aligned, either due to limited data or architecture, limiting their generalizability to represent real-world charts. Instruction-tuning a strongly aligned base VLM can capture the intricacies among diverse chart elements and corresponding text more efficiently. We hypothesize that formulating a simple approach addressing these drawbacks can lead to an effective foundation model capable of complex chart understanding and reasoning in the wild.

We propose ChartGemma, an instruction-tuned multimodal model for chart understanding and reasoning. ChartGemma uses instruction-tuning data for chart representation learning that is directly generated from the chart images, capturing more diverse and relevant information while preserving complex visual features. This also enables us to utilize a much broader array of charts available across the web as we are not restricted by the availability of underlying data tables. ChartGemma develops over PaliGemma (Chen et al., 2023) which has been trained on a much larger alignment dataset. Since ChartGemma uses PaliGemma as its backbone, it is also much smaller than existing chart understanding models, making it suitable for real-world applications. We evaluate ChartGemma across 5 benchmarks spanning chart summarization, question answering, and fact-checking, obtaining state-of-the-art results compared to existing methods. Our qualitative studies also demonstrate that ChartGemma produces more faithful and realistic summaries of complex charts as compared to other methods. Through our elaborate analysis, we put forward ChartGemma as an effective model capable of understanding and reasoning over real-world charts. Our main contributions are:

- We present ChartGemma, a first-of-its-kind multimodal model instruction-tuned for chart understanding and reasoning using data directly generated from chart images.
- ChartGemma utilizes a stronger backbone model and more representative instruction-tuning data, rendering it effective in tackling existing benchmarks across chart summarization, question answering, and fact-checking while being significantly smaller than its counterparts.
- Our extensive quantitative and qualitative studies reveal that ChartGemma generates more faithful and human-like summaries and is extremely capable in understanding and representing complex real-world charts in the wild.

2 Related Work

Chart Representation Learning Chart understanding models initially were either fine-tuned from language or vision-language models (Masry et al., 2022b; Masry and Hoque, 2021; Lee et al.,

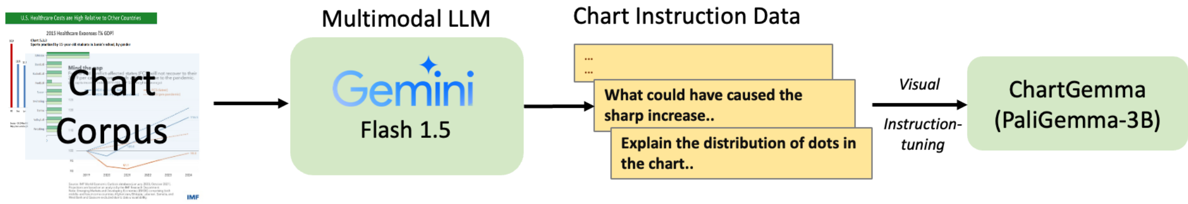


Figure 1: The instruction-tuning data generation process. Chart images are input into Gemini Flash 1.5, which generates visual chart instructions used to fine-tune our model, ChartGemma (please refer to § 3).

2022), or pre-trained using chart-specific learning objectives (Masry et al., 2023; Liu et al., 2022). Recently, instruction-tuning of pre-trained VLMs has been explored for enhancing the general applicability to charts (Meng et al., 2024; Han et al., 2023; Masry et al., 2024; Liu et al., 2023a). Though these methods use diverse sources across the web and synthetic charts for generating instruction-tuning data, they utilize the underlying data table of the charts. Moreover, they train weakly-aligned backbone VLLMs, which often underperform on chart understanding benchmarks due to a lack of specific training and alignment for chart understanding (Kim and Seo, 2024; Kim et al., 2023; Hu et al., 2024; Zhang et al., 2024).

Chart Modeling Benchmarks With charts being the standard medium for data visualization and data-driven decision making, diverse benchmarks have been proposed to evaluate the abilities of LLMs and VLMs on chart understanding. These benchmarks range from close-ended tasks such as question answering (Methani et al., 2020; Masry et al., 2022a) to open-ended generation such as explanation generation in OpenCQA (Kantharaj et al., 2022) and summarization (Shankar et al., 2022). Chart-specific benchmarks evaluate the ability of models to convert charts into data tables (Choi et al., 2019; Masry et al., 2023) or evaluate claims against given data as a part of general multimodal fact-checking benchmarks (Akhtar et al., 2023a,c).

Instruction-tuning across modalities and for charts Instruction-tuning was proposed to generalize the abilities of language models across multiple tasks (Mishra et al., 2022) and has become a common practice for adapting pre-trained LLMs to real-world applications(Alpaca, 2023; Chiang et al., 2023; Ouyang et al., 2022). The success of instruction-tuning for text has led to its adoption as a standard process for multimodal VLMs too (Li et al., 2023; Zhu et al., 2023; Dai et al., 2023). Recently, domain-specific instruction-tuning has been attempted for charts that requires specially curated instruction-tuning data (Han et al., 2023; Masry et al., 2024; Meng et al., 2024). These methods use the underlying data tables of the chart to synthesize the instruction-tuning data. Since the data tables of charts are not capable of capturing the nuance details of charts, especially for real-world charts with complex elements, the instruction-tuning data generated using the data tables is not adequate for training models to be adept at understanding these diverse real-world charts.

3 Chart Instruction Data Generation

This section outlines the details of generating our dataset. We start by curating a diverse chart corpus that encompasses a range of visual styles (§ 3.1), and then use it to generate the visual instruction-tuning data directly from the charts (§ 3.2).

3.1 Assembling the Chart Corpus

Our chart corpus is assembled using a combination of various sources across three categories: (i) Synthetically generated charts from sources such as PlotQA (Methani et al., 2020), (ii) Curated charts from specialized websites such as Statista which typically exhibit limited visual diversity, and (iii) In-the-wild charts harvested from the broader web, such as WebCharts (Masry et al., 2024), noted for their extensive stylistic variety. While prior approaches used accompanying metadata (e.g., titles, data tables, annotations) to generate instructions from LLMs (Han et al., 2023; Meng et al., 2024), our method exclusively utilizes the chart images themselves for generating instruction-tuning data. This approach also allows us to bypass the constraints imposed by metadata availability. In total, our corpus consists of 122,857 chart images. We provide an elaborate breakdown of the chart source and the statistics across each category in Table 4.

3.2 Visual Chart Instructions

We use chart images directly from the above assembled corpus to generate visual instruction-tuning data. This enables us to synthesize data that can train a model to capture not just point information, but complex trends and relations among the chart elements. Following Masry et al. (2024), we generate data across two categories: (i) predefined tasks, which align with common real-world scenarios and benchmarks, and (ii) open-ended tasks. For predefined tasks, we generate data for the following:

1. Chain-of-thought (CoT) involves prompting the model with complex reasoning questions and enhances the visual reasoning capabilities of the model by guiding it through the problem-solving process in a structured manner.

2. Summarization involves prompting the model to generate summaries that succinctly capture the key insights and trends from a chart image to effectively communicate the primary data narratives.

3. Fact Checking asks the model to determine whether stated facts are supported or refuted by the data presented in a chart image. Alongside data generated from our corpus, we use the training sets of existing chart fact-checking tasks (Akhtar et al., 2023a,c) in our instruction-tuning data.

4. Chart-to-Markdown tasks the model with generating the underlying data tables from a chart image in Markdown format. This approach simplifies rendering and parsing the tables, enhancing their accessibility and usability.

5. Program Aided Design (Gao et al., 2022) requires the model to generate executable code that performs necessary calculations and outputs the final answer, delegating complex and challenging mathematical operations to the code interpreter. Alongside synthetic data generated from our corpus, we use the Multimodal LLM to create executable codes for questions in the training split of the ChartQA dataset (Masry et al., 2022b), augmenting our instruction-tuning data with human-written questions and their corresponding code.

Open-ended Tasks We enrich our instruction-tuning data by prompting the Multimodal LLM to generate a variety of tasks typical in real-world scenarios. This approach enhances the generalizability of our models and extends their applicability to diverse real-world settings. Example open-ended tasks include justifying temporal or time-series based trends observed in the chart, describing the different visual elements such as lines, colors, and legends represented by the chart, critically analyzing and comparing visual information, etc. We present concrete examples in § B.2.

We use Gemini Flash-1.5 (Team et al., 2023) due to its robust multimodal performance, cost-effectiveness, and high API rate limits.

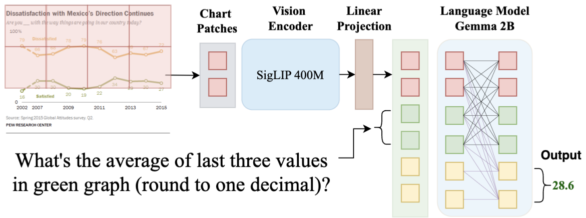


Figure 2: ChartGemma architecture with the SigLIP vision encoder and Gemma-2B language model. Visual tokens (red), prefix tokens (green), and suffix tokens (yellow) interact via full attention (black lines) and causal attention for autoregressive suffix generation (purple lines).

3.3 Key Dataset Characteristics

To underscore the distinct innovations of our dataset relative to prior works, we examine two critical elements: the visual attributes and the quality of the chart instructions.

Visual Attributes Our instruction-tuning dataset features a wide range of instructions that emphasize the visual attributes of chart images. As illustrated in Fig. 5 in Appendix B.2, the examples highlight various visual elements such as lines, shapes, colors, trends, chart types, and positions, all of which are frequently referenced in real-world scenarios.

Quality To demonstrate the strength of our approach in generating high-quality and accurate instructions, we evaluated 100 randomly sampled synthesized instructions. We found that our instructions accurately reflected the chart content in 82% of the cases, which is a significant improvement over the 61% accuracy reported for the ChartInstruct dataset (Masry et al., 2024). Additionally, we observed 8% partially correct answers, similar to that as reported by ChartInstruct. We attribute this improvement in quality to our method's reliance on the chart images, rather than using automatically generated and often erroneous data tables.

4 Modeling and Methodology

4.1 Architecture

ChartGemma uses PaliGemma (Chen et al., 2023) as the backbone architecture, as shown in Fig. 2. The input image is taken in 448x448 resolution and divided into 14x14 pixel patches, each of which is fed into the vision encoder as a separate token. The outputs from the vision encoder are passed through a linear layer that maps the visual features into the LLM embedding space. These visual tokens are then concatenated with the input text embeddings and passed to Gemma-2B. Unlike most previous VLLMs (Li et al., 2023) that indiscriminately apply a causal mask on all image and text tokens, Gemma-2B applies full attention over the input visual and text tokens while a causal mask is applied on the output tokens. This improves the contextual understanding of the image particularly for representing complex relationships among objects. We believe this property provides further advantages when learning representations for chart images containing numerous nuanced complexities.

4.2 Training Setup

Existing chart VLLMs (Meng et al., 2024) typically employ a two-stage training approach that requires an initial step to align the vision encoder and the LLM for understanding chart features, followed by instruction-tuning. In contrast, we only use a single-stage approach where we directly finetune the backbone model on our instruction-tuning data. We believe that the first stage is required by current methods as the VLLM backbones are aligned using a limited amount of image-text pairs with restricted styles and diversity. In contrast, our backbone, PaliGemma, has been trained end-to-end on 10 billion image-text pairs covering a wide variety of styles. This makes our model more adaptable and generalizable to different real-world images (e.g., charts, infographics, documents). We freeze the vision encoder and only finetune the LLM during instruction-tuning. This helps in reducing the computational complexity and also improves training stability given the small batch size used for instruction-tuning PaliGemma.

5 Experiments, Results, and Analyses

5.1 Experimental Setup

We compare ChartGemma against ten baselines comprising of open-source chart-specialist models and VLLMs instruction-tuned on chart data, as well as state-of-the-art closed source multimodal LLMs. Furthermore, we evaluate on a diverse set of 5 established benchmarks evaluating chart representation and reasoning abilities. Further details about the baselines, benchmarks, and evaluation metrics are provided in Appendix C.1

5.2 Performance on closed-ended tasks

We compare the performance of ChartGemma to the various baselines on the closed-ended tasks, namely ChartQA and ChartFC, and present the results in Table 3. We see that Chart VLLMs are generally the better performing set of models compared to specialist chart models. Within Chart VLLMs, we observe that ChartGemma performs the best on ChartQA in terms of the average overall performance and on both the synthetic ChartFC and real-world-based ChartCheck test splits. Particularly, the performance improvements on ChartCheck when using ChartGemma, which is a zero-shot evaluation, can be attributed to the fact that our instruction-tuning dataset is specifically designed to generalize to more realistic charts encountered in this particular evaluation. We observe that it is also powerful for its small size of 3 billion parameters, and only lags in performance to the 13 billion parameter ChartAssistant on the augmented set of ChartQA. The significant improvement of ChartGemma over ChartAssistant on the human-generated split of ChartQA indicates better generalization abilities in understanding more realistic instructions for complex charts.

Given the state-of-the-art performance of ChartGemma, we next perform a series of ablations to test our hypothesis on the criticality of having (i) an instruction-tuning dataset derived from chart images rather than the underlying data tables, and (ii) the importance of a strong backbone model.

Effect of the instruction-tuning data To validate the effectiveness of synthesizing instruction-tuning data directly using the chart images as compared to using their underlying data tables, we compare ChartGemma with a version of PaliGemma instruction-tuned on the dataset presented in ChartInstruct (Masry et al., 2024), which was generated using the chart data tables. We present the results in Table 3. We observe remarkable improvements when using our instruction-tuning data compared to the data proposed by ChartInstruct. The improvements are stark on the human split of ChartQA, indicating that ChartGemma is very efficient in following real-world human instructions. The significantly weak performance of ChartGemma when using the dataset from ChartInstruct is in-line with the observations of the author mentioning a low (61 %) accuracy of the synthetically generated instruction-tuning data (Masry et al., 2024).

Effect of the backbone model We probe the effect of using PaliGemma as the backbone model for ChartGemma, which has better image-text alignment compared to other VLMs, on the downstream performance. We follow existing works (Han et al., 2023; Masry et al., 2024) that use LLaVA (Liu et al., 2023b) as a backbone and train LLaVA-1

Model	#Params	ChartQA (*Relaxed Accuracy*) aug.	human	avg.	Chart Fact Checking (*Accuracy*) ChartFC	ChartCheck T1	ChartCheck T2
Specialist Chart Models							
ChartBERT (Akhtar et al., 2023a)	-	-	-	-	63.8	-	-
Pix2Struct (Lee et al., 2022)	282M	81.6	30.5	56.0	-	-	-
Matcha(Liu et al., 2022)	282M	90.2	38.2	64.2	-	62.80	61.40
UniChart (Masry et al., 2023)	201M	88.56	43.92	66.24	-	-	-
Closed VLMMs							
Gemini Pro (Team et al., 2023)	-	-	-	74.1	65.8	-	-
GPT4-V (OpenAI, 2023)	-	-	-	78.5	69.6	-	-
Chart VLLMs							
ChartLlama (Han et al., 2023)	13B	90.36	48.96	69.66	-	-	-
ChartAssisstant (Meng et al., 2024)	13B	93.90	65.90	79.90	-	-	-
ChartInstruct-Llama2 (Masry et al., 2024)	7B	87.76	45.52	66.64	69.57	70.11	68.80
ChartInstruct-Flan-T5-XL (Masry et al., 2024)	3B	85.04	43.36	64.20	70.27	72.03	73.80
ChartGemma (Ours)	3B	90.80	69.52	80.16	70.33	71.50	74.31

Table 2: Performance on closed-ended generation benchmarks: ChartQA, ChartFC, and ChartCheck. ChartGemma generally outperforms or matches the performance of all the baselines, while being significantly smaller than them (refer to § 5.2).

Model	ChartQA (*Relaxed Accuracy*) aug.	human	avg.	Chart Fact Checking (*Accuracy*) ChartFC	ChartCheck T1	ChartCheck T2
PaliGemma	-	-	71.36	58.26	67.34	68.50
PaliGemma+ChartInstruct	70.24	33.84	52.04	48.58	54.21	51.78
LLaVA+Our dataset	61.12	51.12	56.12	61.28	70.22	70.03
ChartGemma (Ours)	89.44	64.80	77.12	69.95	72.03	73.80

Table 3: Ablation results validating our hypothesis on the effect of our instruction-tuning data and backbone model on downstream tasks (refer to § 5.2).

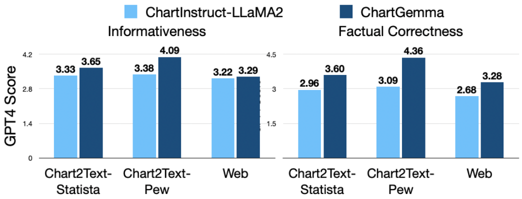


Figure 3: GPT-4 scores (1-5 scale) for the informativeness and factual correctness of outputs from ChartInstruct-LLaMA2 and ChartGemma.

with our instruction-tuning data. We compare this variant (LLaVA+Our dataset) with ChartGemma in Table 3 and observe that ChartGemma performs significantly better as compared to using LLaVA as our backbone. This validates our hypothesis that initializing our architecture with a strongly aligned model leads to better char understanding, reasoning, and generalization capabilities.

5.3 Performance on open-ended tasks

We next compare ChartGemma's performance with baselines on chart understanding open-ended generation benchmarks, OpenCQA (Kantharaj et al., 2022), Chart2Text (Shankar et al., 2022), and our curated 'Web' set. We do not use the BLEU (Papineni et al., 2002) scores for comparison as done by previous works, due to the numerous criticisms of it as an indicative metric (Callison-Burch et al., 2006; Smith et al., 2016) and follow the widespread practice of using strong LLMs as a judge due to their high agreement with human annotators (Zheng et al., 2023). We use GPT4 to evaluate the informativeness and factual correctness of the outputs generated by the models and present the scores in Fig. 3 (refer to the extended results in Appendix C.3). We see that the outputs generated by ChartGemma are generally scored higher as compared to ChartInstruct. We particularly see significant improvement in the factual correctness of the outputs of ChartGemma, probably due to the fact that our instruction-tuning data synthesized using the chart images captures more complex visual elements and PaliGemma being strongly aligned leads to better understanding and reasoning over the charts. Our findings overall indicate that ChartGemma is able to produce more informative outputs while also being factually correct in terms of long-form answering or summarization for the charts.

5.4 Human Evaluation on Summarization

This study investigates the performance of ChartGemma compared to ChartInstruct-LLaMA2 for chart understanding tasks, validated through both human evaluation and GPT-4-based assessment. Human annotators rated summaries generated by both models based on informativeness, factual correctness, and structure, with the results showing ChartGemma consistently outperforming or matching ChartInstruct-LLaMA2 across all metrics. ChartGemma's superior performance, particularly in informativeness and factual accuracy, is attributed to its training on data from chart images, allowing it to capture high-level trends and chart-specific concepts. The study confirms ChartGemma's effectiveness for real-world chart reasoning. More details are provided in Appendix C.4.

5.5 Error Analysis and Challenges

We analyzed the outputs of ChartGemma to understand the shortcomings and areas for improvement and discovered the following patterns of errors.

High Resolution Charts Charts with very large, often skewed dimensions, present challenges for our model which uses an input resolution of 448x448. Resizing these large images can cause written text to become unreadable, leading to errors in the predicted labels and numerical values as depicted in Fig. 13. Although PaliGemma offers a variant supporting up to 896x896 input resolution, it operates significantly slower than the 448x448 version, making it impractical for use on consumer-level GPUs.

Coding Errors While ChartGemma demonstrated state-of-the-art performance on the ChartQA benchmark, excelling in complex numerical reasoning and compositional questions, it occasionally generates erroneous code that cannot be executed. As depicted in Fig. 13, the model sometimes refers to undeclared variables within the code. We believe that integrating an LLM with enhanced coding capabilities could further improve our performance on the ChartQA benchmark.

Charts with Complex Visual Styles Although our instruction-tuning corpus predominantly features real-world charts from the broad web, ChartGemma tends to exhibit lower factual correctness and informativeness when evaluated on these charts compared to those from specialized websites like Pew or Statista, which have less visual diversity. This disparity, illustrated in Fig. 3, highlights the need for further enhancements to improve the generalizability of chart understanding models across various visual styles.

6 Conclusion and Future Work

In the landscape of rising excitement for chart understanding and reasoning models and methods, we present ChartGemma, a multimodal model instruction-tuned on data generated directly from a diverse range of real-world chart images using a state-of-the-art backbone architecture. ChartGemma addresses two crucial shortcomings of existing instruction-tuned chart models: the instruction-tuning data is generated from the underlying data tables instead of the chart images, limiting their adaptability and extendibility to real-world, and use weakly aligned backbone models, restricting their generalizability. Our simple approach yields significant improvements over existing chart representation models, with a relatively smaller model in terms of number of parameters. Our extensive error analyses and human studies show that ChartGemma produces more realistic, informative, and factually correct outputs as compared to its contemporaries.

As future work, we aim to formulate a more diverse instruction-tuning dataset which is created using human written instructions capturing varied nuances present in charts. We also aim to propose a more generalized benchmark catered to addressing complex visual elements in charts with more chart relevant evaluation metrics.

Limitations

Despite the effectiveness of our instruction-tuning approach and our model, there are notable limitations. Firstly, the instruction-tuning data is generated using a proprietary LLM, which could restrict the model's use in certain commercial environments. Secondly, the input resolution of our model's vision encoder is capped at 448x448; any increase in resolution leads to a quadratic rise in processing time. Third, we depend on the closed-source model, GPT4, for evaluating crucial metrics such as Informativeness and Factual Correctness. The frequent updates and potential deprecation of closed-source models pose challenges for the reproducibility of our results. Lastly, the model is prone to hallucinations, occasionally producing factually incorrect statements or erroneous code. We advise users to implement robust guardrails and exercise caution when deploying our model in real-world applications.

Ethics Statement

Since our model generates responses autoregressively, it is prone to errors and hallucinations. The outputs can sometimes be misleading or contain inaccuracies. Additionally, there is no guarantee that the codes generated by our model will be free from malicious content. Therefore, it is crucial for users of our model to implement strict safety guidelines to mitigate these potential risks.

The authors and their research collaborators conducted the human evaluation study, so there was no monetary compensation. Moreover, the samples are randomly shuffled to prevent any bias towards our model's responses. Finally, there were no personal identification information collected during this study.

All models employed in our experiments are publicly available and licensed for research use. Furthermore, all chart images in our dataset were sourced from existing, publicly available research papers that have filtered out any offensive content. Finally, we plan to release our instruction-tuning dataset along with the model for research purposes.

Acknowledgement

We would like to thank the anonymous reviewers for their helpful feedback. This research was supported by the Natural Sciences Engineering Research Council (NSERC) of Canada and Canada Foundation for Innovation (CFI).

References

Mubashara Akhtar, Oana Cocarascu, and Elena Simperl. 2023a. Reading and reasoning over chart images for evidence-based automated fact-checking. *arXiv preprint arXiv:2301.11843*.

Mubashara Akhtar, Oana Cocarascu, and Elena Simperl. 2023b. Reading and reasoning over chart images for evidence-based automated fact-checking. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 399–414, Dubrovnik, Croatia. Association for Computational Linguistics.

Mubashara Akhtar, Nikesh Subedi, Vivek Gupta, Sahar Tahmasebi, Oana Cocarascu, and Elena Simperl. 2023c. Chartcheck: An evidence-based fact-checking dataset over real-world chart images. *arXiv preprint arXiv:2311.07453*.

Alpaca. 2023. Alpaca. `https://crfm.stanford.edu/2023/03/13/alpaca.html`.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluating the role of Bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 249–256, Trento, Italy. Association for Computational Linguistics.

Xi Chen, Xiao Wang, Lucas Beyer, Alexander Kolesnikov, Jialin Wu, Paul Voigtlaender, Basil Mustafa, Sebastian Goodman, Ibrahim Alabdulmohsin, Piotr Padlewski, Daniel Salz, Xi Xiong, Daniel Vlasic, Filip Pavetic, Keran Rong, Tianli Yu, Daniel Keysers, Xiaohua Zhai, and Radu Soricut. 2023. Pali-3 vision language models: Smaller, faster, stronger. *Preprint*, arXiv:2310.09199.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

J. Choi, Sanghun Jung, Deok Gun Park, J. Choo, and N. Elmqvist. 2019. Visualizing for the non-visual: Enabling the visually impaired to use visualization. *Computer Graphics Forum*, 38.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Preprint*, arXiv:2305.06500.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2022. Pal: Program-aided language models. *arXiv preprint arXiv:2211.10435*.

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*.

Enamul Hoque, Parsa Kavehzadeh, and Ahmed Masry. 2022. Chart question answering: State of the art and future directions. *Journal of Computer Graphics Forum (Proc. EuroVis)*, pages 555–572.

Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Chen Li, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. 2024. mplug-docowl 1.5: Unified structure learning for ocr-free document understanding. *Preprint*, arXiv:2403.12895.

Shankar Kantharaj, Xuan Long Do, Rixie Tiffany Ko Leong, Jia Qing Tan, Enamul Hoque, and Shafiq Joty. 2022. Opencqa: Open-ended question answering with charts. In *Proceedings of EMNLP (to appear)*.

Geewook Kim, Hodong Lee, Daehee Kim, Haeji Jung, Sanghee Park, Yoonsik Kim, Sangdoo Yun, Taeho Kil, Bado Lee, and Seunghyun Park. 2023. Visually-situated natural language understanding with contrastive reading model and frozen large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11989–12010, Singapore. Association for Computational Linguistics.

Geewook Kim and Minjoon Seo. 2024. On efficient language and vision assistants for visually-situated natural language understanding: What matters in reading and reasoning. *Preprint*, arXiv:2406.11823.

Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. 2022. Pix2struct: Screenshot parsing as pretraining for visual language understanding. *arXiv preprint arXiv:2210.03347*.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2023. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *arXiv preprint arXiv:2306.00890*.

Fangyu Liu, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Yasemin Altun, Nigel Collier, and Julian Martin Eisenschlos. 2022. 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*.

Ahmed Masry and Enamul Hoque. 2021. Integrating image data extraction and table parsing methods for chart question answering. *Chart Question Answering Workshop, in conjunction with the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–5.

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. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (to appear)*. Association for Computational Linguistics.

Ahmed Masry, Do Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022a. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, Dublin, Ireland. Association for Computational Linguistics.

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

Ahmed Masry, Mehrad Shahmohammadi, Md Rizwan Parvez, Enamul Hoque, and Shafiq Joty. 2024. Chartinstruct: Instruction tuning for chart comprehension and reasoning. *Preprint*, arXiv:2403.09028.

Fanqing Meng, Wenqi Shao, Quanfeng Lu, Peng Gao, Kaipeng Zhang, Yu Qiao, and Ping Luo. 2024. Chartassisstant: A universal chart multimodal language model via chart-to-table pre-training and multitask instruction tuning. *arXiv preprint arXiv:2401.02384*.

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 (WACV)*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 Technical Report. *Preprint*, arXiv:2303.08774.

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

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Kantharaj Shankar, Leong Rixie Tiffany Ko, Lin Xiang, Masry Ahmed, Thakkar Megh, Hoque Enamul, and Joty Shafiq. 2022. Chart-to-text: A large-scale benchmark for chart summarization. In *In Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), 2022*.

Aaron Smith, Christian Hardmeier, and Joerg Tiedemann. 2016. Climbing mont BLEU: The strange world of reachable high-BLEU translations. In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 269–281.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, and Jiahui Yu et al. 2023. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Yanzhe Zhang, Ruiyi Zhang, Jiuxiang Gu, Yufan Zhou, Nedim Lipka, Diyi Yang, and Tong Sun. 2024. Llavar: Enhanced visual instruction tuning for text-rich image understanding. *Preprint*, arXiv:2306.17107.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

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 Related Work

Chart Representation Learning Chart understanding models initially were either fine-tuned from language or vision-language models (Masry et al., 2022b; Masry and Hoque, 2021; Lee et al., 2022), or pre-trained using chart-specific learning objectives (Masry et al., 2023; Liu et al., 2022). Recently, instruction-tuning of pre-trained VLMs has been explored for enhancing the general applicability to charts (Meng et al., 2024; Han et al., 2023; Masry et al., 2024; Liu et al., 2023a). Though these methods use diverse sources across the web and synthetic charts for generating instruction-tuning data, they utilize the underlying data table of the charts and train a weakly-aligned backbone VLM.

Chart Modeling Benchmarks With charts being the standard medium for data visualization and data-driven decision making, diverse benchmarks have been proposed to evaluate the abilities of LLMs and VLMs on chart understanding. These benchmarks range from close-ended tasks such as question answering (Methani et al., 2020; Masry et al., 2022a) to open-ended generation such as explanation generation in OpenCQA (Kantharaj et al., 2022) and summarization (Shankar et al., 2022). Chart-specific benchmarks evaluate the ability of models to convert charts into data tables (Choi et al., 2019; Masry et al., 2023) or evaluate claims against given data as a part of general multimodal fact-checking benchmarks (Akhtar et al., 2023a,c).

Instruction-tuning across modalities and for charts Instruction-tuning was proposed to generalize the abilities of language models across multiple tasks (Mishra et al., 2022) and has become a common practice for adapting pre-trained LLMs to real-world applications(Alpaca, 2023; Chiang et al., 2023; Ouyang et al., 2022). The success of instruction-tuning for text has led to its adoption as a standard process for multimodal VLMs too (Li et al., 2023; Zhu et al., 2023; Dai et al., 2023). Recently, domain-specific instruction-tuning has been attempted for charts that requires specially curated instruction-tuning data (Han et al., 2023; Masry et al., 2024; Meng et al., 2024). These methods use the underlying data tables of the chart to synthesize the instruction-tuning data. Since the data tables of charts are not capable of capturing the nuance details of charts, especially for real-world charts with complex elements, the instruction-tuning data generated using the data tables is not adequate for training models to be adept at understanding these diverse real-world charts.

B Chart Instruction Data Generation

B.1 Chart Corpora Collection

We collect chart across 3 categories based on their source and method of generation as mentioned in § 3.1. We show the exact statistics and sources under each category in Table 4.

Sources for instruction-tuning tasks For the pre-defined tasks used for generating instruction-tuning data, we also augment the instructions generated by the multimodal LLM with the training sets of existing benchmark datasets.

B.2 Instruction Dataset Analysis

Our instruction-tuning dataset comprises of both closed-ended response generation and open-ended answering. Fig. 4 shows diverse visual instruction-tuning tasks that are generally inspired from existing chart evaluation benchmarks, and Fig. 5 shows diverse visual instruction-tuning tasks inspired from open-ended chart understanding and reasoning.

Instruction-tuning dataset quality As mentioned in § 3.3, our instruction-tuning dataset's instructions accurately reflect the chart content approximately 82% of the times, and are partially correct 8% times. We present some examples where our instructions are correct and incorrect in Table 5 and partially correct in Table 6.

B.3 Prompt Templates for Instruction-tuning Data Generation

We present the prompt templates provided to Gemini Flash-1.5 to generate instruction-tuning data for the program-aided design task in Fig. 6 and an open-ended task in Fig. 7. Our prompt templates draw inspiration from the templates used in ChartInstruct (Masry et al., 2024) and the ChartQA prompt used in Gemini Flash (Team et al., 2023).

C Experiments and Results

C.1 Experimental Setup

Baselines We compare ChartGemma against baselines comprising of open-source chart-specialist models and VLLMs instruction-tuned on chart data, as well as state-of-the-art closed source multimodal LLMs. Chart-specialist models include *ChartBERT* (Akhtar et al.,

Dataset	Predefined Tasks					Open Ended Tasks					
	CoT Reasoning	Chart Summarization	Fact Checking	Chart-to Markdown	Coding Abilities	Trend Analysis	Data Comparison	Data Interpretation	Data Visualization	Others	#Charts
Synthetic Sources											
PlotQA	-	-	-	5000	-	-	-	-	-	-	5000
ChartFC	-	-	28000	-	-	-	-	-	-	-	12702
Specialized Websites											
Statista	2688	4996	1296	2377	42098	334	172	373	231	3027	19748
Pew	11951	4999	1251	1784	10034	281	290	307	129	2873	7401
OECD	243	500	644	20838	357	39	47	69	31	489	21712
OWID	717	500	375	2285	1490	40	38	61	28	547	3803
ChartCheck (Wikipedia)	-	1527	7603	-	-	98	96	178	65	1642	1530
General Web											
WebCharts	10576	50046	6434	18216	3400	4331	6283	4680	1785	51436	50961
Total	26,175	62,241	45,603	22,603	57,379	792	6926	988	2269	60,014	122,857

Table 4: The number of generated examples for each tasks based on data samples of the mentioned dataset. Some of the charts are used in multiple tasks. In the last column, we show the number of distinct charts used for instruction generation samples.

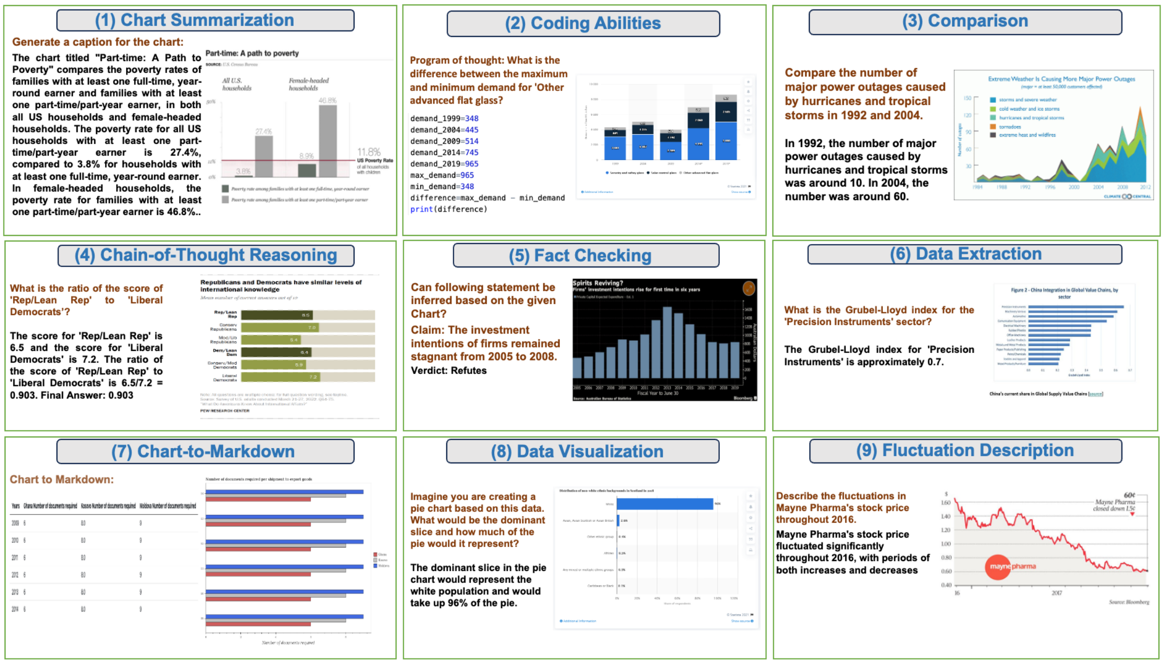


Figure 4: Diverse examples from our visual instruction-tuning tasks that focuses on the visual attributes of the chart images which are highlighted in green.

2023c), *Pix2Struct* (Lee et al., 2022), *MatCha* (Liu et al., 2022), and *UniChart* (Masry et al., 2023). Chart VLLMs include *ChartLlaMA* (Han et al., 2023), *ChartAssistant* (Meng et al., 2024), and *ChartInstruct*'s (Masry et al., 2024) two variants with LLaMA2 and Flan-T5-XL. We also compare ChartGemma against two closed-source multimodal LLMs, namely Gemini Pro (Team et al., 2023) and GPT4-V (OpenAI, 2023).

Downstream Tasks We evaluate ChartGemma on a diverse set of 5 established benchmarks evaluating chart representation and reasoning abilities: (i) ChartQA (Masry et al., 2022b) – a factoid chart question answering dataset, (ii) ChartFC (Akhtar et al., 2023a) and (iii) ChartCheck (Akhtar et al., 2023b) – chart fact checking datasets, (iv) OpenCQA (Kantharaj et al., 2022) – an open-ended chart question answering dataset, and (v) Chart2Text (Shankar et al., 2022) – a chart summarization dataset. While ChartQA and ChartFC focus on closed-ended generation, OpenCQA and Chart2Text evaluate open-ended generation abilities of the models. We also manually curate a set of 100 charts downloaded from the web completely unseen by any model. We refer to this set as 'Web' in our results, and use them for comparing the summarization ability of the models.

Evaluation Metrics Following existing works, we use relaxed accuracy (RA) for ChartQA, accuracy for ChartFC, and use GPT4 as a judge for open-ended generation tasks, i.e. Chart2Text, OpenCQA, and our curated Web set of charts and

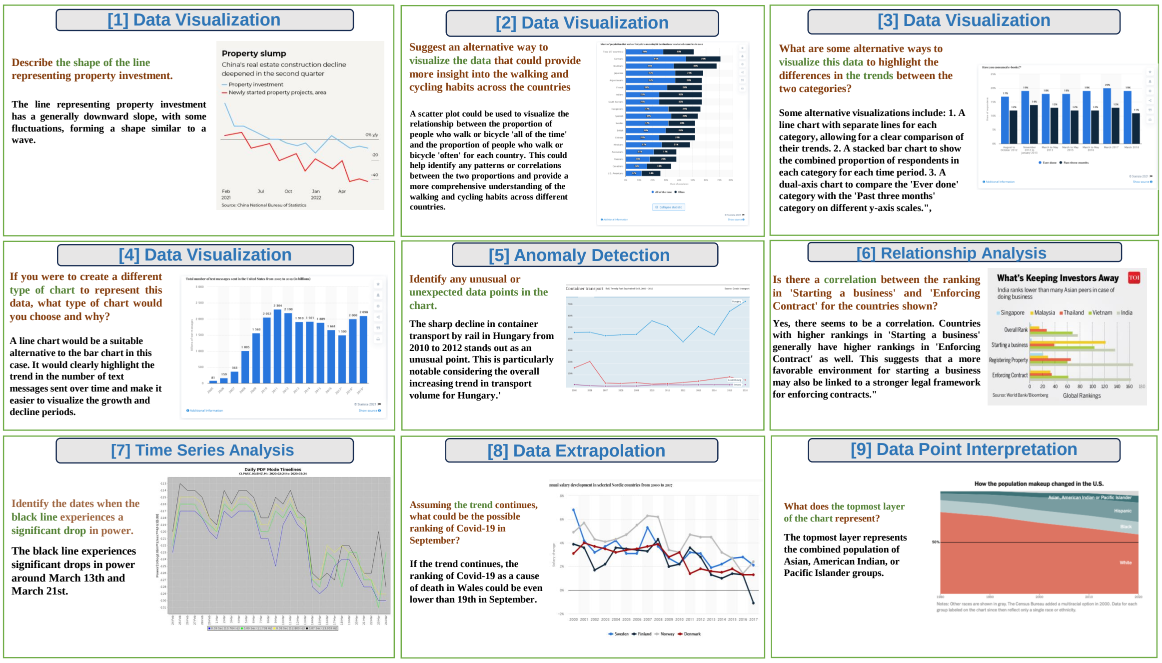


Figure 5: Diverse examples from our open-ended instruction-tuning tasks that focuses on the visual attributes of the chart images which are highlighted in green.

measure the informativeness and factual correctness on a scale of 1-5 (Post, 2018).

To ensure the reproducibility of our work, we present the hyperparameters settings for instruction-tuning and fine-tuning on the benchmarks in Table 7. All experiments were conducted on a 4 A100 GPUs (80GB) machine using the JAX framework[2].

C.2 Prompt templates for evaluation

We show the prompt given to GPT4 for evaluating the outputs of the open-ended tasks, Chart2Text and our curated 'Web' set for summarization and OpenCQA in Fig. 8 and Fig. 9, respectively.

C.3 GPT4 evaluation on open-ended generation tasks

We show the informativeness, factual correctness, and relevance results on the open-ended generation tasks, namely Chart2Text(Statista and Pew), OpenCQA, and our curated 'Web' set of charts in Table 8.

C.4 Human Evaluation Study

Though using online LLMs like GPT4 as a judge has been shown to have a high correlation with human annotation (Zheng et al., 2023), there haven't been studies on measuring this correlation explicitly for chart understanding tasks. Hence, to ensure our observations, evaluations, and conclusions are robust, we perform a human study on the manually curated set of 100 charts, 'Web'. Similar to GPT4 evaluation, we compare the informativeness, factual correctness, and structure of the outputs generated by ChartGemma with ChartInstruct-LLaMA2.

We first use ChartInstruct-LLaMA2 and ChartGemma to generate summaries for these samples in the Web set. We then ask 2 different annotators to rate all the responses based on the above metrics (informativeness, factual correctness, structure) from 1-5 (5 being the highest) so we can also measure agreement between the annotations[3]. We present the outputs randomly to the annotators to prevent any biases towards the models and present the evaluation results in Fig. 11.

From Fig. 11, we observe that ChartGemma consistently outperforms or matches ChartInstruct-LLaMA2 on all the metrics, and the findings are in-line with those observed when using GPT4 for evaluation (Section 5.3). We observe that ChartGemma is equally well structured, yet is more informative and significantly more factually correct. Better informativeness probably stems from the fact that ChartGemma is trained on data generated from the chart images and not just the underlying data tables, enabling it to learn high level trends and concepts specific to charts. Furthermore, our instruction-

[2]https://github.com/google/jax

[3]We found a Cohen's Kappa of 0.538 for the agreement.

Example Prompt - Generate Instruction-tuning data for Program-Aided Design

```
Generate numerical and visual question-answer pairs for an LLM that we are trying to tune for Chart Numerical and Visual
Reasoning. Your response should be in a json format where each example has three fields: input: which only asks a
numerical/visual question, program of thought: a python program that can be executed to produce the final answer, and
final answer: which is the final answer to the input question based on the chart image.
For the final answer X, follow the following instructions:
* X should contain as few words as possible.
* Don't paraphrase or reformat the text you see in the image.
* If the final answer has two or more items, provide it in the list format like [1, 2].
* When asked to give a ratio, give out the decimal value like 0.25 instead of 1:4.
* When asked to give a percentage, give out the whole value like 17 instead of decimal
like 0.17%.
* Don't include any units in the answer.
* Try to include the full label from the graph when asked about an entity.
Generate ten questions that contain some numerical operations such as, but not limited to, max, min, sum, average,
difference, ratio, median, mode, ..etc. Generate another five questions that not only have numerical operations, but also
some visual aspects such as leftmost, rightmost, top, bottom, middle, peak, colors, ..etc. Generate five simple data
retrieval questions that ask about values, x-labels, or legend labels from the chart. Generate another five yes/no
numerical reasoning questions whose answers must be either Yes or No. Generate another four questions that ask to count
some elements in the chart (e.g., the number of bars/pie slices/colors/x-labels).
Remember that the program of thought must be an executable python code that solves the question step by step and prints
the answer in the end.
```

Figure 6: Prompt to generate instruction-tuning data for the program-aided design task using Gemini Flash-1.5.

Example Prompt - Generate Instruction-tuning data for Open-ended Tasks

```
Generate different instruction-tuning tasks for an LLM that we are trying to tune for Chart Understanding. Your response
should be in a json format where each example has three fields: task type, input: which only asks a question or an
instruction related to the task type and the given chart, and expected output: which is the answer to the input
question/instruction based on the input information. Use the following chart image to generate 10 unique tasks
```

Figure 7: Prompt to generate instruction-tuning data for open-ended tasks using Gemini Flash-1.5.

Example Prompt - Evaluating generated summaries

```
You will be provided with two summaries generated by different models for chart summarization.
Your task is to evaluate each summary based on three key factors:

Informativeness: How much useful and relevant information from the chart does the summary cover? Does it effectively
convey the main trends and insights?
Factual Correctness: How accurate is the summary in reflecting the information presented in the chart?
Structure: How well-structured is the summary? Does it include an introduction, a body with key insights, and a
conclusion?
You are required to assign a score from 1 to 5 for each factor, for each summary. Please provide your ratings in the
following JSON format:
{
    'summary 1': {
        'Informativeness' : score,
        'Factual Correctness' : score,
        'Structure' : score,
    },
    'summary 2': {
        'Informativeness' : score,
        'Factual Correctness' : score,
        'Structure' : score,
    },
}
Do not return anything else other than the json above.
```

Figure 8: Example prompt to evaluate open-ended summary generation for Chart2Text and the 'Web' set of charts using GPT4.

Example Prompt - Evaluating OpenCQA

```
You will be provided with two answers generated by different models for a question about a chart image.
Your task is to evaluate each answer based on three key factors:
Informativeness: How much useful and relevant information from the chart does the answer cover?
Factual Correctness: How accurate is the answer in reflecting the information presented in the chart?
Relevance: How relevant is the answer to the given question?

You are required to assign a score from 1 to 5 for each factor, for each answer. Please provide your ratings in the
following JSON format:
{
    'summary 1': {
        'Informativeness' : score,
        'Factual Correctness' : score,
        'Relevance' : score,
    },
    'summary 2': {
        'Informativeness' : score,
        'Factual Correctness' : score,
        'Relevance' : score,
    },
}}
```

Figure 9: Example prompt to evaluate open-ended answer generation for OpenCQA using GPT4.

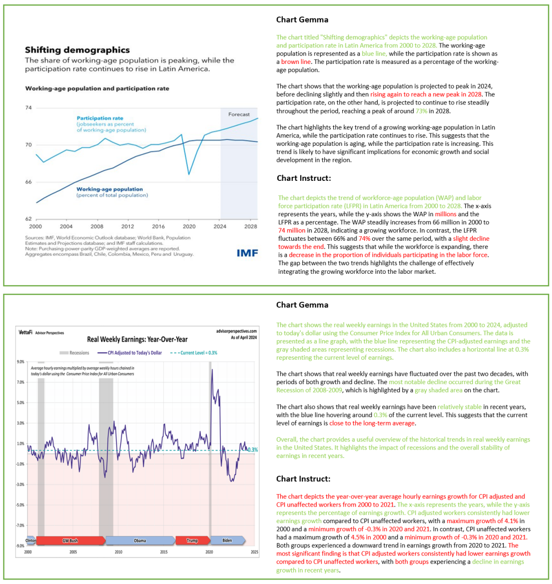


Figure 10: Comparison between ChartGemma and ChartInstruct-LLama2 for chart captioning.

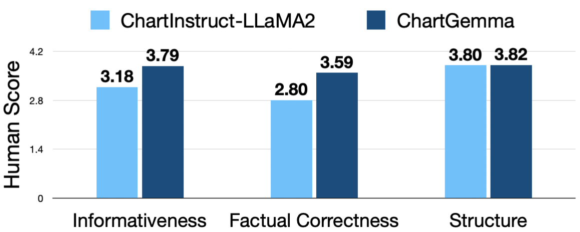


Figure 11: Human evaluation scores on the informativeness, factual correctness, and structure of outputs generated by ChartInstruct-LLaMA2 and ChartGemma.

tuning data and a strong backbone model promote capturing more complex visual elements of charts, leading to more factual correctness. Overall, since our evaluation is performed on charts sampled randomly in the wild from the web, ChartGemma's strong performance validates its effectiveness as a strong candidate in understanding and reasoning over real-world charts.

During the human evaluation study, we provided the human annotators with the same instructions used to prompt GPT4 as depicted in Fig. 8 and Fig. 9. We recruited two human volunteers for the study from our research lab, both were of South-east Asian (Indian subcontinent) origin and adept in the English language.

We show the results of human evaluation when measuring the informativeness, factual correctness, and structure of outputs generated by ChartInstruct-LLaMA2 and ChartGemma on the 'Web' set of charts scraped from the web in Table 9. We see that ChartGemma significantly outperforms ChartInstruct-LLaMA2 in terms of informativeness and factual correctness and they match in the structure of the generated summary.

C.5 Error Analysis

Fig. 13 show typoes and coding errors produced by our model.

C.6 Convergence of ChartGemma

We probe the learning dynamics of ChartGemma by checking the downstream accuracy with the number of instruction-tuning epochs and present the trends in Fig. 12. We interestingly observe that ChartGemma converges very quickly, with the best performance observed at epoch 2. We attribute this characteristic to the strong alignment of PaliGemma rendering it effective in adapting to our relatively generalizable instruction-tuning dataset. This indicates that PaliGemma is a very

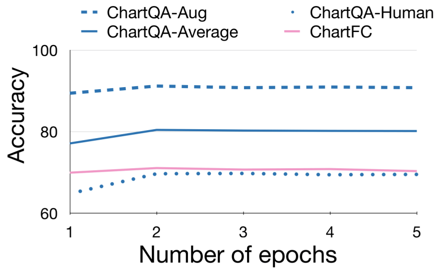


Figure 12: Effect of the number of epochs on instruction-tuning ChartGemma. We observe very quick convergence during training (refer to § C.6). For ChartQA, accuracy is relaxed accuracy (§ 5.1).

efficient backbone for visual instruction-tuning of chart data, and might generalize when trained with a much larger number of samples as well. We leave this exploration as future work.

C.7 Sample Outputs from ChartGemma

In Fig. 14, we provide some sample outputs on various tasks.

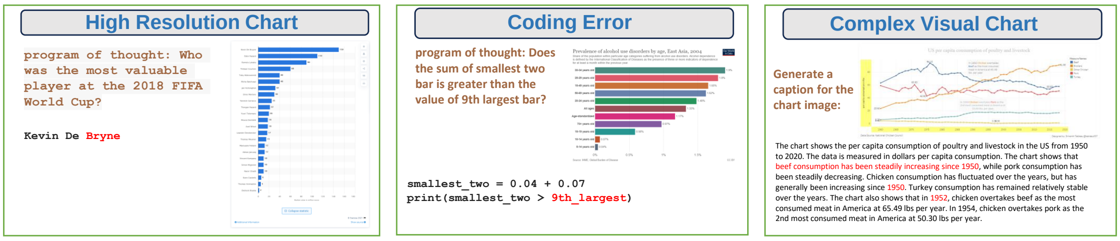


Figure 13: Some samples that our model, ChartGemma, has typos, coding errors, and factually incorrect statements in their outputs. The errors are shown in red.

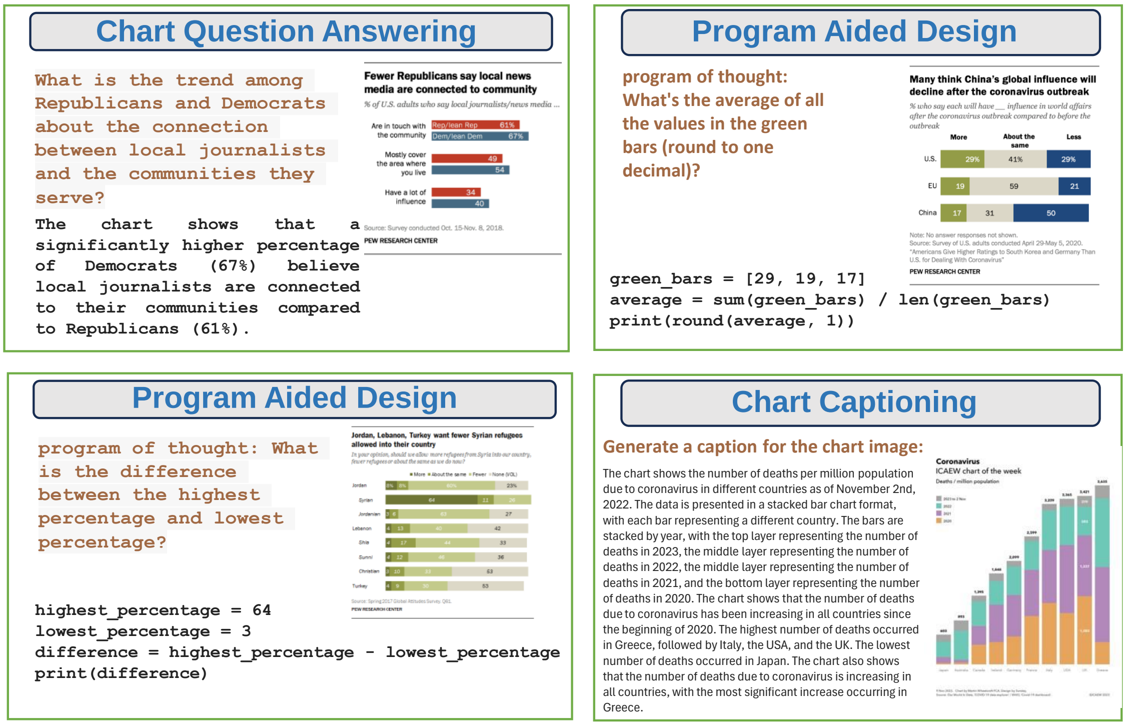


Figure 14: Sample outputs generated by ChartGemma on various downstream tasks.

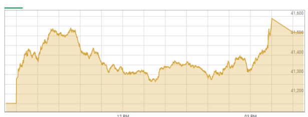
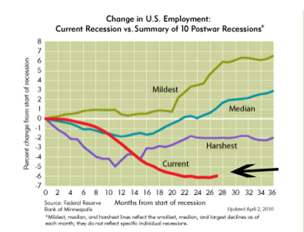
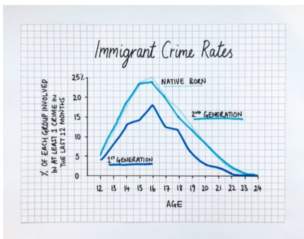
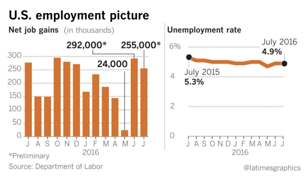
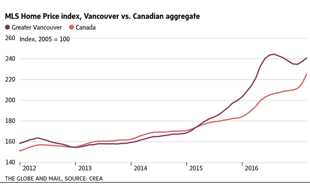
Chart Image	Chart Instructions	Answers	Label
	Based on the chart, what can you conclude about the performance of the British Pound against the US Dollar in the last six months?	The British Pound has depreciated against the US Dollar over the past six months, with a significant drop between May and August and a slight recovery towards the end of the period.	Correct
	Based on the trend observed in the chart, what would you predict the value of the data to be at 4:00 PM?	Based on the trend, the value of the data at 4:00 PM would likely be slightly lower than the value at 3:00 PM, possibly around 41,400.	Correct
	Based on the trend of the 'Current' recession line, what is the predicted percent change from the start of the recession after 35 months?	It is difficult to predict the exact percent change after 35 months based solely on the trend of the 'Current' recession line. However, based on the upward trend after 28 months, it is likely to be around -5%.	Correct
	Based on the chart, what might be a possible explanation for the high crime rate among native-born individuals?	It is difficult to say for sure based on this chart alone. However, some potential explanations could be socioeconomic factors, access to resources, or cultural influences.	Correct
	Compare the market cap of Deutsche Bank and Credit Suisse in 2021.	Deutsche Bank and Credit Suisse in 2021. In 2021, Deutsche Bank's market cap was significantly higher than Credit Suisse's. Deutsche Bank's market cap was around 25 billion while Credit Suisse's was around 15 billion.	Incorrect
	Describe the trend of job gains in the chart.	The trend of job gains is generally increasing, with a few dips in between. It experienced a significant peak in July 2016 and May 2016.	Incorrect
	Compare the home price index in Greater Vancouver and Canada in 2015.	In 2015, the home price index in Greater Vancouver was significantly higher than in Canada. While Greater Vancouver's index was around 180, Canada's index was around 170.	Incorrect

Table 5: Example answers generated from Gemini Flash 1.5 based on the instructions supplied. We present some correct generations and some incorrect generations (with highlights in red).

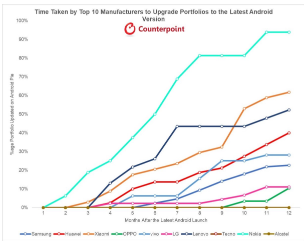
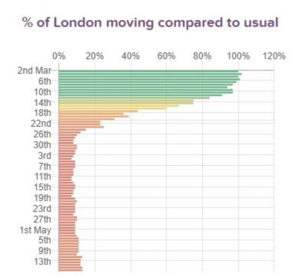
Chart Image	Chart Instructions	Answers	Label
	What percentage of Samsung's portfolio was updated to the latest Android version after 9 months?	Around 25% of Samsung's portfolio was updated to the latest Android version after 9 months.	Partially correct
	Identify the dates when the percentage of Londoners moving compared to usual was below 30	The dates when the percentage was below 30% are: March 3rd, March 7th, March 11th, March 15th, March 19th, March 23rd, March 27th, 1st May, 5th May, 9th May, and 13th May. Missing dates	Partially correct

Table 6: Example answers generated from Gemini Flash 1.5 based on the instructions supplied. We present some partially correct generations here (with highlights in orange).

Experiment	# Epochs	Learning Rate	Batch Size	Hours
	Instruction-tuning			
ChartGemma	5	5e-5	32	58
	Ablations			
PaliGemma (chartinstruct)	1	5e-5	32	22
LLaVA + our dataset	1	2e-5	32	11
ChartGemma	1	5e-5	32	11
	Finetuning on benchmarks			
PaliGemma (ChartFC)	10	5e-5	32	2
PaliGemma (ChartCheck)	10	5e-5	32	4
ChartInstruct-LLama2 (ChartCheck)	10	2e-5	32	2
ChartInstruct-Flan-T5-XL (ChartCheck)	10	2e-5	32	1

Table 7: Hyperparameters and training details of our experiments.

	Informativeness	Factual Correctness	Structure
Statista			
ChartInstruct-LLama2	3.33	2.96	3.58
ChartGemma	3.65	3.60	3.66
Pew			
ChartInstruct-LLama2	3.38	3.09	3.65
ChartGemma	4.09	4.36	3.85
OpenCQA			
ChartInstruct-LLama2	3.54	3.46	4.56
ChartGemma	3.26	3.48	4.19
Web			
ChartInstruct-LLama2	3.22	2.68	3.33
ChartGemma	3.29	3.28	3.76

Table 8: GPT4 scores (from 1-5, with 5 being the highest) on the informativeness and factual correctness of outputs generated by ChartInstruct-LLaMA2 and ChartGemma (refer to § 5.3).

	Informativeness	Factual Correctness	Structure
ChartInstruct-LLaMA2	3.18	2.80	3.80
ChartGemma	3.79	3.59	3.82
$p-value$	6.31×10^{-6}	2.68×10^{-7}	0.457

Table 9: Human evaluation scores on the informativeness, factual correctness, and structure of outputs generated by ChartInstruct-LLaMA2 and ChartGemma. We also provide the p-values by performing Mann-Whitney U Tests.