ChartAgent: A Modular Agentic Framework for Accurate Chart-to-Table Extraction with Visual Zooming

Anonymous ACL submission

Abstract

Extracting structured tables from chart images is a challenging task that underpins numerous downstream document analysis applications. While previous studies have demonstrated that multimodal large language models (MLLMs) and vision-language models (VLMs) can convert charts into tables, these models frequently fail to adhere to strict formatting standards, omit fine-grained labels, or introduce numerical inaccuracies. In this work, we introduce ChartAgent, a plug-and-play, agent-based framework that augments any off-the-shelf VLM through a two-stage agentic pipeline. In the first stage, a chart-to-table pretrained VLM generates an initial table directly from the chart image. In the second stage, a ReAct LLM-based agent iteratively corrects the generated table by cross-verifying visual regions and textual entries. This agent can optionally utilize a novel zooming tool designed for detailed and precise inspection of complex, densely packed chart areas. To evaluate the effectiveness of ChartAgent, we benchmarked its performance on the ChartQA dataset against state-of-the-art methods. Our experiments demonstrate consistent improvements over both VLM-only and single-pass correction baselines across structural and numerical metrics. The modular design of ChartAgent enables seamless integration with any VLM without requiring additional fine-tuning. This approach significantly enhances header alignment, numerical fidelity, and overall table quality, providing a robust and efficient solution for accurate chart-to-table extraction.

1 Introduction

Charts are everywhere from scientific papers and technical reports to financial statements and business presentations and play a key role in sharing numbers and trends (Huang et al., 2024).

These graphical tools transform raw datasets into intuitive visual patterns, making complex information immediately accessible and serving as the foundation for effective communication, strategic decision-making, and scholarly inquiry.

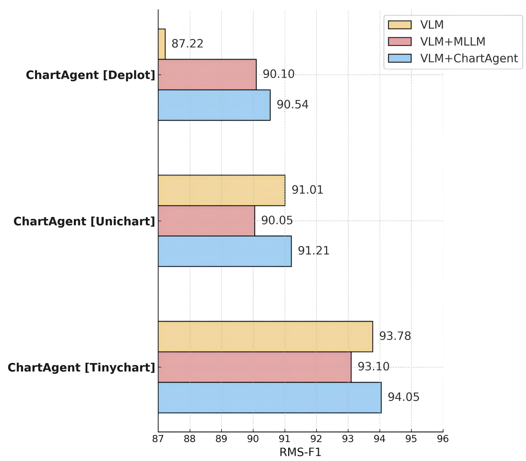


Figure 1: AgentChart performance on various VLM and compared to VLM + MLLM models for the chart-to-table extraction on RMS–F_1 metric, showing that AgentChart achieves the highest score.

Yet the data inside these charts often stays trapped as an image, making it hard to run analyses, write automated reports, or answer questions. Automated extraction of chart images into structured tables is essential for quantitative information embedded in charts. Such chart-to-table extraction enables tasks like data analysis, report generation, and question answering over document collections. Although recent multimodal models can interpret a wide range of chart types, but One-shot generation often presents various shortcomings, such as adhering to precise table schemas and can misread small labels or crowded legends. These limitations hinder reliable data extraction in real-world settings.

VLMs (Masry et al., 2023), (Liu et al., 2022), (Zhang et al., 2024a)(Meng et al., 2024) have achieved strong performance on standard benchmarks by converting chart visuals into linearized table representations. However, their one-shot output may contain swapped headers, merged cells, or incorrect numerical values when faced with diverse chart designs and They often make mistakes when addressing numerical calculation questions (Meng et al., 2024), which require reasoning steps for accurate answers. Single-pass correction with a general large language model can fix some errors but lacks the granularity needed to address fine-grained mistakes under strict formatting constraints.

To overcome these challenges, we propose ChartAgent, A modular pipeline that integrates an agentic correction stage with any existing chart-to-table VLM. In the first stage, the VLM produces an initial table from the chart image. In the second stage, a ReAct LLM-based Agent (Yao et al., 2023) iteratively refines both structure and content: it detects missing rows, swapped headers, and misread values through visual-textual cross-checking, and applies corrective edits. One major contribution of our work is our Zoom tool facilitates this process by partitioning the chart into overlapping quadrants for high-resolution inspection of complex areas or areas with high uncertainty. By combining an initial draft extraction with iterative focused correction, ChartAgent substantially reduces residual errors without retraining the underlying models.

We performed extensive evaluation on the ChartQA (Masry et al., 2022) dataset, showing that it outperforms the VLM-only and VLM + MLLM baselines in three complementary metrics: relative number set similarity (RNSS) proposed by (Masry et al., 2022) based on the graphIE metric proposed in (Luo et al., 2021), Relative Mapping Similarity (RMS-F1) proposed by (Liu et al., 2022), and Relative Distance (RD-F1) proposed by (Kim et al., 2024). An ablation study confirms the importance of the agentic workflow and selective zooming, and qualitative examples highlight the system's ability to recover missing labels and split merged segments. ChartAgent thus offers a robust and extensible solution for accurate chart-to-table extraction in diverse applications and our main contributions are :

1. We introduce AgentChart, a modular agent-based correction pipeline that augments existing vision-language models for chart-to-table extraction without retraining.
2. We propose a Zoom Tool that enables fine-grained visual inspection of crowded chart regions, significantly improving label and value recovery.
3. We validate AgentChart on the ChartQA benchmark, achieving state-of-the-art performance across three complementary metrics, and demonstrate the effectiveness of agentic correction via ablation and qualitative studies.

2 RELATED WORK

2.1 General Purpose LLM

Multimodal large language models (MLLMs) have demonstrated promising results in initial evaluations on chart-to-table tasks. These models either closed or open source can interpret chart images and convert them into structured tabular data without the need for task specific fine tuning. Examples of closed sources includes Claude sonnet or Gemini and open source like InternLM-XComposer (Zhang et al., 2024b) and LLAMA (Touvron et al., 2023) that achieved promising scores on chart related tasks. While these MLLMs provide a scalable and flexible alternative to dedicated chart models, allowing broad application across diverse document types and reducing the need for extensive fine-tuning on charts However, these models often struggle with chart-to-table tasks that require strict formatting constraints. Despite strong general capabilities, they may not reliably follow precise table schemas specified via prompt and not always give precise numerical values.

2.2 Multimodal chart understanding models

Vision- large language models (VLM) (Du et al., 2022) are widely used for chart-related tasks and, more specifically, for chart-to-table extraction. UniChart (Masry et al., 2023) is pretrained on a large corpus of charts covering diverse topics and visual styles, leveraging a Donut (Kim et al., 2022) based vision encoder and a chart-grounded text decoder to optimize low-level element extraction and high-level reasoning tasks before fine-tuning on chart-to-table parsing, which yields state-of-the-art performance on multiple extraction benchmarks; however, its reliance on a chart-specific pretraining corpus may limit robustness to novel chart formats beyond those seen during pretraining. DePlot (Liu et al., 2022) employs a Pix2Struct (Lee et al., 2023) derived image-to-text Transformer trained end-to-end on a standardized plot-to-table task, converting chart images into linearized markdown tables that can be directly prompted to an MLLM, though it has a limited performance on highly stylized or unconventionally formatted charts outside its training distribution. ChartAssistant (Meng et al., 2024) introduces a two-stage training pipeline via ChartSFT's chart-text pairs first pretraining on chart-to-table translation to align visual elements with structured text, then multitask instruction tuning across QA, summarization, and reasoning offering two variants (260 M-parameter Donut-based and 13 B-parameter SPHINX-based (Lin et al., 2023)) that outperform UniChart and ChartLlama (Han et al., 2023) under zero-shot real-world settings; nonetheless, the 13 B-parameter variant's inference demands and potential missing on chart types absent from ChartSFT present deployment and generalization challenges. Finally, TinyChart (Zhang et al., 2024a) distills efficient chart-to-table capabilities into a 3 B-parameter MLLM by integrating Visual Token Merging to compress high-resolution inputs and a Program-of-Thoughts learning strategy to generate executable Python code for numerical calculations, achieving state-of-the-art results on ChartQA (Masry et al., 2022), Chart-to-Text, and Chart-to-Table benchmarks with two time faster inference; however, its PoT synthesis step may introduce additional latency and it may struggle to strictly adhere to complex table schemas when such constraints are prescribed in prompts. While these approaches contribute valuable insights into chart-to-table extraction, persistent challenges such as adhering to strict table formatting, managing variability in chart layouts, highlight the need for further methodological refinements in chart-related vision-language modeling.

2.3 Agentic Workflows in chart related tasks

Agentic workflows and AI agents have led to substantial gains in the autonomy and adaptability of MLLM systems, enabling them to perceive, reason, and act within complex environments. These agents facilitate the development of AI systems capable of dynamic decision-making and task execution, thereby enhancing the efficiency and scalability of LLM-powered systems. In chart-related tasks, existing implementations have predominantly focused on auxiliary functions, such as identifying chart regions or converting data into visual formats. For instance, ChartCitor (Goswami et al., 2025) employs a multi-agent framework to provide fine-grained visual attributions in chart question-answering scenarios, enhancing the explainability of AI-generated responses. Similarly, METAL (Li et al., 2025) utilizes a multi-agent approach for chart generation, decomposing the task into specialized agents that collaboratively produce high-quality charts. Despite these advancements, the deployment of agentic frameworks in chart-to-table extraction tasks remains underexplored. This process involves extracting structured tabular data from complex chart images, a task that poses significant challenges due to the variability in chart designs and the intricacies of visual encoding. Our approach introduces a plug-and-play agentic framework that actively intervenes in the chart-to-table pipeline. By deploying specialized agents to identify and correct errors made by chart-to-table-specific VLMs, we enhance the accuracy and reliability of the extracted tabular data. This agentic intervention enables dynamic error detection and correction, allowing the system to adapt to diverse chart formats and reduce the propagation of inaccuracies in downstream tasks. Such an approach not only improves the fidelity of data extraction but also contributes to the development of more robust chart understanding AI systems.

3 Methodology

3.1 ChartAgent Architecture

Figure 2 illustrates the summary of our proposed ChartAgent as a plug-and-play pipeline that enhances any chart-to-table VLM, such as TinyChart (Zhang et al., 2024a) or UniChart (Masry et al., 2023), by layering a correction LLM agent on top of its output. The core workflow unfolds in two stages.

Stage 1, A pretrained chart-to-table VLM takes the chart image and output an initial structured table. these models are good at reading overall layouts and most numbers and labels, but they can sometimes miss small text or give some numerical errors when charts are crowded.

Stage 2, An LLM-based ReAct agent is invoked that both reasons about and acts upon the VLM's preliminary table. The agent takes as input the original chart image plus the raw table, then iterate in order to : (i) refines its structure by detecting missing rows, swapped headers, or unintended merged cells through visual and textual cross-checking using the zoom tool; (ii) verifies content by targeting specific chart regions to correct missing or misread

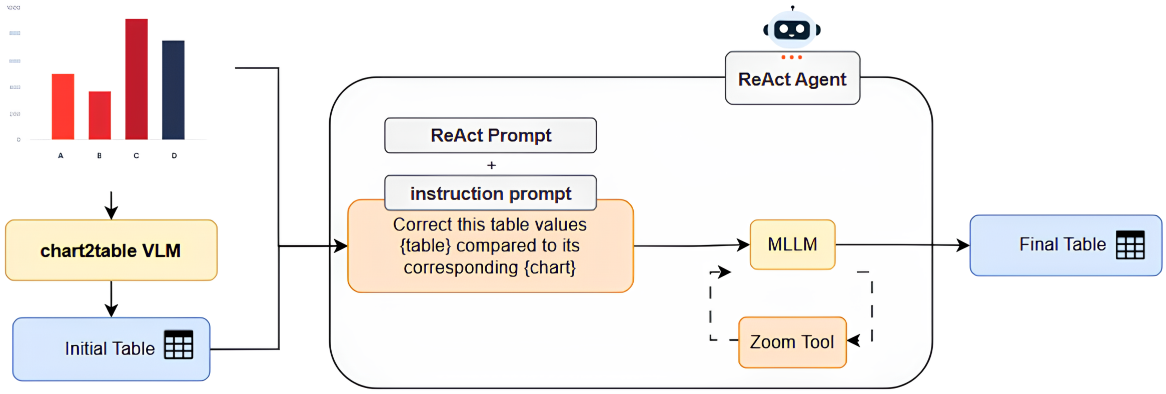


Figure 2: Overview of ChartAgent. The chart image is provided to the VLM, which outputs an initial table. This table, along with the chart, the ReAct prompt, and the instruction prompt, are given as inputs to the agent. The agent then iteratively refines the table, optionally using the zoom tool and accessing the message history, until it either reaches a final output table that it considers correct or hits the iteration limit.

numerical/textual entries by using the zoom tool; and (iii) applies edits on the input table by correcting the textual, numerical values or adding missing information if needed. By combining extraction with focused correction, ChartAgent overcomes residual errors and leverages the strengths of both systems without retraining large models.

Algorithm 1 ChartAgent Algorithm

Require: Image I, Prompt P, VLM, MLLM, Zoom_Tool $\mathcal{T}$

$T_0 \leftarrow$ VLM.generate_table(I)
$A_0 \leftarrow$ MLLM.answer(P, T_0)
history $\leftarrow [(P, A_0)]$
for $k = 1$ to MaxSteps **do**
  $E_k \leftarrow$ MLLM.answer$(history)$
  history.append(E_k)
  if $E_k ==$ `Correct` **then**
    return Final Answer T_{k-1}
  else if $E_k ==$ `Zoom` **then**
    $crop \leftarrow \mathcal{T}$.execute$(I)$
    history.append$(crop)$
    continue {Next iteration with refined view}
  end if
end for
$T_k \leftarrow$ MLLM.answer$(history)$
return Final Answer T_k

3.2 Zoom Tool

The zoom tool is a tool that we developed to enable fine-grained inspection as shown in the Figure 3, which transform the chart to a higher resolution by uniformly upscale it by a factor using Lanczos interpolation and partitions it into four quadrants labeled (upper left, upper right, lower left, and lower right) for selective access. When the agent encounters crowded tick labels or dense legends, it requests the appropriate quadrant rather than processing the entire image, thereby isolating the appropriate region of interest. This targeted zooming leverages the same logic as Multimodal CoT Prompting. By iterating between selective zoom and table refinement, ChartAgent ensure that even the smallest chart details are correctly transcribed. Our results show that this tool significantly helps in improving performance by enabling targeted and fine-grained inspection, thus ensuring accurate transcription of even the smallest chart details.

4 Experimentation and Results

4.1 Implementation Details

Our AgentChart system is implemented as a two-stage, plug-and-play pipeline. In the first stage, a vision-language model (VLM) performs initial chart-to-table extraction. In the second stage, a ReAct-based agent powered by a large language model (LLM) iteratively refines the output table through structured reasoning and visual-textual cross-verification.

Stage 1: Chart-to-Table Extraction. We evaluated three state-of-the-art VLMs DePlot (Liu et al., 2022), UniChart (Masry et al., 2023), and TinyChart (Zhang et al., 2024a) to gen-

Figure 3: Overview of the Zoom Tool.The LLM agent select the zoom tool and provide as an argument in the tool call which quadrant is needed.

erate initial tables from chart images. Each model was used without any modifications to its published configuration.

Stage 2: Agentic Refinement. For the correction phase, we built ReAct-style agents using Anthropic Claude Sonnet 3.5 MLLM. this agent iteratively inspect and edit the initial tables by reasoning over both the raw chart image and the extracted table.

Zoom Tool : To support fine-grained inspection of densely populated or ambiguous chart regions, we developed a custom Zoom Tool. This lightweight image-processing module dynamically crops the chart into four labeled quadrants (upper-left, upper-right, lower-left, and lower-right), allowing the agent to selectively inspect specific areas without reprocessing the full image.

4.2 Baselines Methods

ChartAgent's performance was evaluated against two baseline approaches under consistent experimental settings:

1. VLM-Only: The chart-to-table models DePlot, UniChart, and TinyChart were run independently, producing raw tables without any additional correction or refinement.
2. Single-Pass MLLM Correction: A general-purpose large language model was applied once to post-process the VLM output, without iterative reasoning or visual cross-checking.
3. ChartAgent (Ours): Our full pipeline augments the VLM output using a multi-step, agent-driven correction stage that incorporates structured reasoning and targeted visual inspection via the Zoom Tool.

4.3 Benchmark Dataset

All evaluations were conducted using the ChartQA dataset (Masry et al., 2022), a widely adopted benchmark for chart-to-table extraction. It includes a diverse collection of real-world bar, line, and pie charts, each paired with a ground-truth table in markdown format. ChartQA is known for its visual diversity and annotation quality, making it a robust and challenging testbed. To ensure fair and reproducible comparisons, all results are reported on the held-out test split of the dataset, following standard practice in prior work.

Following prior chart-to-table works, we evaluate extracted tables using three scores that capture different aspects of the generated table quality .

4.4 Evaluation Metrics

Relative Number Set Similarity (RNSS)

RNSS measures how well the unordered multiset of numeric entries in the predicted table matches the ground truth. Let

$$P = \{p_i\}_{i=1}^{N}, \quad T = \{t_j\}_{j=1}^{M}$$

be the sets of predicted and true values. First define the relative distance

$$D(p,t) = \min\Big(1, \frac{|p-t|}{|t|}\Big). \qquad (1)$$

Method	#Parameters	RMS–F_1	RNSS	RD–F_1
UniChart	260M	91.01	94	88
Deplot	1.3B	87.22	95.57	90.91
TinyChart@768	3B	93.78	96.88	91.1
SimPlot	374M	-	-	92.32
Claude Sonnet 3.5	-	90.13	96.67	92.02
TinyChart+ChartAgent (ours)	3B	**94.05**	**97.95**	**94.3**

Table 1: Quantitative results on the ChartQA test set across various chart types, evaluated using RD–F_1, RMS–F_1, and RNSS metrics for chart-to-table extraction. SimPlot results are directly taken from their original paper and report only RD–F1.

We then compute an optimal one-to-one matching $X \in \{0,1\}^{N\times M}$ between P and T. RNSS is given by

$$\text{RNSS} = 1 - \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} X_{ij}\, D(p_i, t_j)}{\max(N, M)}, \quad (2)$$

which ranges from 0 (no overlap) to 1 (perfect match).

Relative Mapping Similarity (RMS)

RMS accounts for both structure and content by comparing full table entries as triples (r, c, v). Let $p_i = (p_i^r, p_i^c, p_i^v)$ and $t_j = (t_j^r, t_j^c, t_j^v)$ denote the row key, column key, and value. We define

$$\text{NL}_\tau(a, b) = \text{normalized Levenshtein distance},$$

and the relative distance as

$$D_\theta(v_p, v_t) = \min\Big(1, \frac{|v_p - v_t|}{|v_t|}\Big).$$

Then the similarity between entries $D_{\tau,\theta}(p_i, t_j)$ is $\big(1 - \text{NL}_\tau(p_i^r \| p_i^c,\ t_j^r \| t_j^c)\big)\ \big(1 - D_\theta(p_i^v, t_j^v)\big)$. Using the same matching X, we compute precision and recall:

$$\text{RMS}_{precision} = 1 - \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} X_{ij}\, D_{\tau,\theta}(p_i, t_j)}{N}, \quad (3)$$

$$\text{RMS}_{recall} = 1 - \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} X_{ij}\, D_{\tau,\theta}(p_i, t_j)}{M}. \quad (4)$$

and report the harmonic mean of the precision and recall as RMS-F_1 .

Relative Deviation (RD)

RD focuses exclusively on numeric fidelity under the established matching X and it is proposed by (Kim et al., 2024). to take into considration the equivalent textual data Using D_θ as above, we define:

$$\text{RD}_{precision} = 1 - \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} X_{ij}\, D_\theta(p_i^v, t_j^v)}{N}, \quad (5)$$

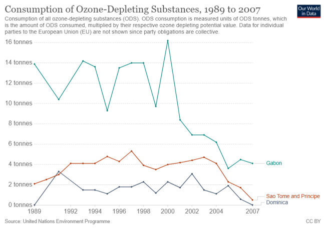
$$\text{RD}_{recall} = 1 - \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} X_{ij}\, D_\theta(p_i^v, t_j^v)}{M}. \quad (6)$$

and combine them via harmonic mean to obtain RD-F_1 .

These three metrics RNSS, RMS-F_1, and RD-F_1 and together provide a thorough, quantitative evaluation of numeric set overlap, full table structure, and raw value accuracy. RNSS measure the overall numeric overlap regardless of position but ignores row/column alignments; RMS-F_1 jointly evaluates structural correspondence and value correctness yet may be sensitive to minor string mismatches in row/column keys; RD-F_1 isolates pure numerical fidelity but does not account for textual alignment in the table. By employing all three, we capture complementary aspects matching, structural alignment, and raw deviation to ensure a comprehensive assessment of chart-to-table extraction quality.

4.5 Main Results

Table 1 reports the quantitative performance of all methods on the ChartQA test set, measured with three key metrics: RNSS, RMS–F_1, and RD–F_1. Our ChartAgent pipeline consistently outperforms

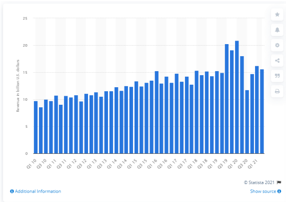


Year	Gabon	São Tomé	Dominica
1989	14.32	2.38	0.28
1992	4.42	-	-
1994	4.42	1.78	14.04
1996	4.62	2.08	14.04
1998	4.25	2.58	14.42
2000	4.32	2.58	16.62
2002	4.72	3.38	7.25
2004	4.42	1.38	6.5
2007	4.42	0.78	0.28

Unichart output (with errors in red)

Year	Gabon	São Tomé	Dominica
1989	14.0	2.0	0.0
1992	10.5	3.0	3.2
1994	14.0	4.0	1.5
1996	14.0	4.0	1.5
1998	14.0	4.0	1.5
2000	16.0	4.5	2.0
2002	8.5	5.0	3.0
2004	7.0	4.0	1.2
2007	4.0	0.5	0.2

Agent-corrected using ChartAgent

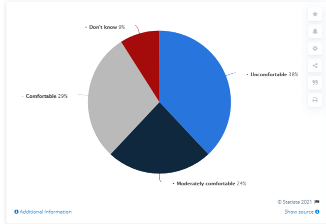


Quarter	Revenue (US$ bn)
...	...
Q11	11.2
Q1 '12	9.7
Q4 '11	10.4
Q3 '11	10.7
Q2 '11	9.0
Q11	10.8
Q4 '10	9.8
Q3 '10	10.0
Q2 '10	8.6

Unichart output (with errors in red)

Quarter	Revenue (US$ bn)
...	...
Q2 '12	11.2
Q1 '12	9.7
Q4 '11	10.4
Q3 '11	10.7
Q2 '11	9.0
Q1 '11	10.8
Q4 '10	9.8
Q3 '10	10.0
Q2 '10	8.6

Agent-corrected using ChartAgent

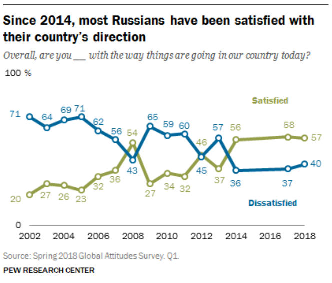


Comfort level	Share of respondents
Uncomfortable	38%
Moderately comfortable	24%
Comfortable	29%
Missing row: Don't know | 9%	

Unichart output (with errors in red)

Comfort level	Share of respondents
Uncomfortable	38%
Moderately comfortable	24%
Comfortable	29%
Don't know	9%

Agent-corrected using ChartAgent

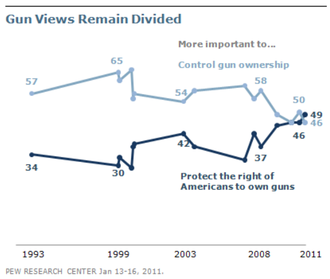


Year	Dissatisfied	Satisfied
2002	0	20
2004	0	26
2006	0	36
2008	0	54
2010	59	0
2012	0	46
2014	36	56
2016	0	58
2018	40	0

Tinychart output (with errors in red)

Year	Dissatisfied	Satisfied
2002	71	20
2004	64	27
2006	69	26
2008	56	32
2010	59	34
2012	60	45
2014	36	57
2016	37	58
2018	40	57

Agent-corrected using ChartAgent

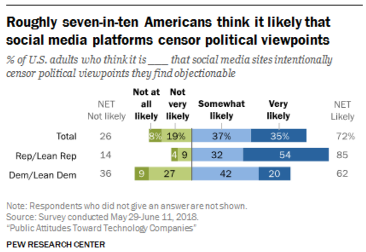


Year	Control gun ownership	Protect the right of Americans to own guns
1993	0	0
1999	0	0
2003	0	0
2008	58	0
2011	50	49

Tinychart output (with errors in red)

Year	Control gun ownership	Protect the right of Americans to own guns
1993	57	34
1999	65	30
2003	54	42
2008	58	37
2011	50	46

Agent-corrected using ChartAgent



Entity	Not at all likely	Not very likely	Somewhat likely	Very likely	NET likely
Dem/Lean Dem	nan	27	42	20.0	62
Rep/Lean Rep	nan	9	32	nan	85
Total	819.0	19	37	35.0	72

Tinychart output (with errors in red)

Entity	Not at all likely	Not very likely	Somewhat likely	Very likely	NET likely
Dem/Lean Dem	9	27	42	20	62
Rep/Lean Rep	4	9	32	54	85
Total	7	19	37	35	72

Agent-corrected using ChartAgent

Figure 4: Examples of chart-to-table extraction and correction using AgentChart on Tinychart@768 (Zhang et al., 2024a) and Unichart (Masry et al., 2023)

both the standalone VLMs and the single-pass VLM+MLLM setup. It also outperforms single MLLM highlighting the effectiveness of the agentic correction stage in enhancing chart-to-table extraction accuracy.

ChartAgent achieves the highest performance across all three evaluation metrics RNSS, RMS–F_1, and RD–F_1, demonstrating superior structural alignment and numerical fidelity compared to both VLM-only and VLM+MLLM baselines. Notably, the agentic correction stage contributes significantly to improvements in header matching (as reflected in RNSS) and raw value accuracy (captured by RD–F_1).

4.5.1 Ablation Study

To isolate the contributions of each component, we conducted ablations by first removing the Zoom

Method	RMS-F_1	RNSS
Unichart	91.01	94
Unichart + (Claude)	90.05	95.2
Unichart + Agent (Claude)	91.21	96
Deplot	87.22	95.57
Deplot + (Claude)	90.1	97.1
Deplot + Agent (Claude)	90.54	97.4
Tinychart	93.78	96.88
Tinychart + (Claude)	93.1	96.91
Tinychart + Agent (Claude)	94.05	97.95

Table 2: Ablation study. We tested the MLLM based and agent based correction on different VLMs.

Tool by using only chart to table model and MLLM and Using diffrant MLLM for the Agent. As we can see in the table 2 Skipping the agentic stage reduces all three metrics substantially, underscoring the value of iterative, tool-enabled corrections.

4.5.2 Qualitative Analysis

Figure 4 presents representative examples where ChartAgent corrects errors made by the base VLM. In a dense bar chart, the agent identifies and restores missing small-value labels; in a pie chart with merged slices, it accurately splits and relabels adjacent segments. These case studies illustrate how targeted zooming and structured reasoning combine to enhance table extraction.

5 Conclusion

In this work, we presented ChartAgent, a flexible, plug-and-play framework that layers an agentic correction stage on top of existing chart-to-table vision-language models. By combining a strong initial extractor (e.g., TinyChart or UniChart) with a React LLM-based agent that iteratively refines both structure and content and by introducing a Zoom Tool for high-resolution inspection ChartAgent achieves significant gains on the ChartQA benchmark. Our experiments that have been conducted on different VLMs and metrics demonstrate consistent improvements in header alignment, numerical fidelity, and overall table quality compared to VLM-only and single-pass correction baselines. Importantly, these gains are obtained without any retraining of large models, making ChartAgent an efficient and extensible solution for accurate chart-to-table extraction.

6 Limitations

Despite its strengths, AgentChart has some limitations. First, the iterative nature of the ReAct LLM-based agent, combined with the Zoom Tool and the two-stage pipeline, introduces additional processing steps that may increase computational cost and latency. This can be a limitation for real-time applications. However, it does not negatively impact offline scenarios such as the ingestion stage in retrieval-augmented generation (RAG), where the system still benefits from iterative refinement.
Besides, we also observed that the performance of ChartAgent can be influenced by the initial table extraction from the vision-language model (VLM). In cases where the VLM output suffers from severe misreads or layout issues, the refinement process may be less effective. In future work, we aim to reduce this dependence and enhance the robustness of the iterative correction process.
Finally, our method shows strong potential for the chart-to-table extraction task, which is the primary focus of this study. Nevertheless, we believe the approach can be extended to other chart-related tasks such as chart question answering, chart-to-text generation, and open-ended chart understanding.

References

Jinyue Chen, Lingyu Kong, Haoran Wei, Chenglong Liu, Zheng Ge, Liang Zhao, Jianjian Sun, Chunrui Han, and Xiangyu Zhang. 2024. Onechart: Purify the chart structural extraction via one auxiliary token. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 147–155.

Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Songyang Zhang, Haodong Duan, Wenwei Zhang, Yining Li, and 1 others. 2024. Internlm-xcomposer2-4khd: A pioneering large vision-language model handling resolutions from 336 pixels to 4k hd. *Advances in Neural Information Processing Systems*, 37:42566–42592.

Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. 2022. A survey of vision-language pre-trained models. *arXiv preprint arXiv:2202.10936*.

Kanika Goswami, Puneet Mathur, Ryan Rossi, and Franck Dernoncourt. 2025. Chartcitor: Multi-agent framework for fine-grained chart visual attribution. *arXiv preprint arXiv:2502.00989*.

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*.

Kung-Hsiang Huang, Hou Pong Chan, Yi R Fung, Haoyi Qiu, Mingyang Zhou, Shafiq Joty, Shih-Fu

Chang, and Heng Ji. 2024. From pixels to insights: A survey on automatic chart understanding in the era of large foundation models. *IEEE Transactions on Knowledge and Data Engineering*.

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.

Wonjoong Kim, Sangwu Park, Yeonjun In, Seokwon Han, and Chanyoung Park. 2024. Simplot: Enhancing chart question answering by distilling essentials. *arXiv preprint arXiv:2405.00021*.

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.

Bingxuan Li, Yiwei Wang, Jiuxiang Gu, Kai-Wei Chang, and Nanyun Peng. 2025. Metal: A multi-agent framework for chart generation with test-time scaling. *arXiv preprint arXiv:2502.17651*.

Ziyi Lin, Chris Liu, Renrui Zhang, Peng Gao, Longtian Qiu, Han Xiao, Han Qiu, Chen Lin, Wenqi Shao, Keqin Chen, and 1 others. 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. 2022. Deplot: One-shot visual language reasoning by plot-to-table translation. *arXiv preprint arXiv:2212.10505*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306.

Junyu Luo, Zekun Li, Jinpeng Wang, and Chin-Yew Lin. 2021. Chartocr: Data extraction from charts images via a deep hybrid framework. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1917–1925.

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*.

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

J OpenAI Achiam, S Adler, S Agarwal, L Ahmad, I Akkaya, FL Aleman, D Almeida, J Altenschmidt, S Altman, S Anadkat, and 1 others. 2023. Gpt-4 technical report. arxiv. *arXiv preprint arXiv:2303.08774*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, and 1 others. 2023. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. 2024. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 13040–13051.

Liang Zhang, Anwen Hu, Haiyang Xu, Ming Yan, Yichen Xu, Qin Jin, Ji Zhang, and Fei Huang. 2024a. Tinychart: Efficient chart understanding with visual token merging and program-of-thoughts learning. *arXiv preprint arXiv:2404.16635*.

Pan Zhang, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Rui Qian, Lin Chen, Qipeng Guo, Haodong Duan, Bin Wang, Linke Ouyang, and 1 others. 2024b. Internlm-xcomposer-2.5: A versatile large vision language model supporting long-contextual input and output. *arXiv preprint arXiv:2407.03320*.