Unmasking Deceptive Visuals: Benchmarking Multimodal Large Language Models on Misleading Chart Question Answering

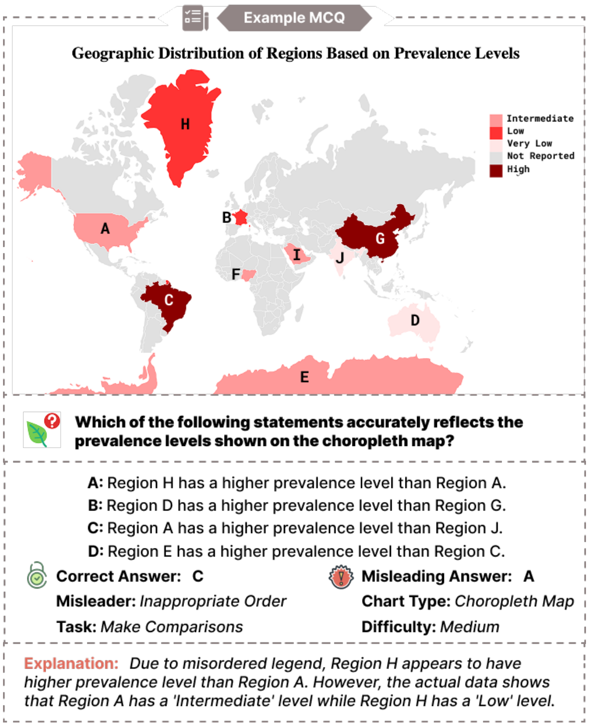
Anonymous ACL submission

Abstract

Misleading visualizations, which manipulate chart representations to support specific claims, can distort perception and lead to incorrect conclusions. Despite decades of research, they remain a widespread issue—posing risks to public understanding and raising safety concerns for AI systems involved in data-driven communication. While recent multimodal large language models (MLLMs) show strong chart comprehension abilities, their capacity to detect and interpret misleading charts remains unexplored. We introduce Misleading ChartQA benchmark, a large-scale multimodal dataset designed to evaluate MLLMs on misleading chart reasoning. It contains 3,026 curated examples spanning 21 misleader types and 10 chart types, each with standardized chart code, CSV data, multiple-choice questions, and labeled explanations, validated through iterative MLLM checks and exhausted expert human review. We benchmark 24 state-of-the-art MLLMs, analyze their performance across misleader types and chart formats, and propose a novel region-aware reasoning pipeline that enhances model accuracy. Our work lays the foundation for developing MLLMs that are robust, trustworthy, and aligned with the demands of responsible visual communication. Code and dataset will be publicly released.

1 Introduction

Misleading visualizations have long posed challenges in chart comprehension and public communication (Tufte and Graves-Morris, 1983). As early as the 1950s, the influential book *How to Lie with Statistics* illustrated how selectively constructed charts could distort data and manipulate public perception (Huff, 2023). Despite decades of awareness, misleading designs remain common today. For example, in 2020, the Georgia Department of Public Health released a COVID-19 bar chart sorted by case count rather than date, falsely implying a decline in infections (McFall-Johnsen, 2020) (fig. 6 A). Another widely recognized example is the standard world map under Mercator Projection (fig. 6 B), which distorts country sizes by exaggerating areas near the poles (Kennedy et al., 2000; O'Brien, 2024). These real-world cases illustrate how charts can subtly mislead audiences, posing risks to public understanding and highlighting the importance of trustworthy data communication.


Example MCQ
Geographic Distribution of Regions Based on Prevalence Levels
Intermediate
Low
Very Low
Not Reported
High
H
B
A
I
J
G
F
C
D
E

Which of the following statements accurately reflects the prevalence levels shown on the choropleth map?

A: Region H has a higher prevalence level than Region A.
B: Region D has a higher prevalence level than Region G.
C: Region A has a higher prevalence level than Region J.
D: Region E has a higher prevalence level than Region C.

Correct Answer: **C**
Misleading Answer: **A**
Misleader: *Inappropriate Order*
Chart Type: *Choropleth Map*
Task: *Make Comparisons*
Difficulty: *Medium*

Explanation: *Due to misordered legend, Region H appears to have higher prevalence level than Region A. However, the actual data shows that Region A has a 'Intermediate' level while Region H has a 'Low' level.*

Figure 1: An example multiple-choice question (MCQ) from our benchmark. Each MCQ includes a misleading chart, a question, multiple answer options, the correct answer and a set of labels. A detailed explanation is also provided to illustrate the chart's misleading aspects.

Recent advances in multimodal large language models (MLLMs) have shown strong performance on chart-related tasks such as question answering (Xia et al., 2024; Masry et al., 2022), captioning (Huang et al., 2023; Rahman et al., 2023), and structure extraction (Chen et al., 2024a). However, most existing work focuses on factual interpretation and overlooks the critical challenge of detecting and reasoning about misleading visual content. Although this issue has long been recognized in the visualization literature (Tufte and Graves-Morris, 1983; Ge et al., 2023), it remains largely unaddressed in the context of MLLMs.

We attribute this gap to three key challenges: (1) the theoretical difficulty of defining and organizing diverse misleader types and aligning them with specific chart formats; (2) the complexity and cognitive effort required to design high-quality question-answer pairs that reflect realistic misleading scenarios; and (3) the substantial amount of expert human labor needed for accurate annotation and validation. As MLLMs are increasingly deployed in high-stakes domains—news summarization, policy analysis, scientific communication—their ability to recognize and resist visual manipulation becomes essential. This capability is not only key to combating misinformation but also to ensuring responsible AI deployment aligned with user intent, legal norms, and societal values.

To address this gap, we present the Misleading ChartQA benchmark, a large-scale multimodal dataset for evaluating MLLMs' ability to identify and reason about misleading charts. Our work builds on theoretical foundations that define common misleading features (misleaders) (Börner et al., 2019; Lo et al., 2022; Lan and Liu, 2024) and multiple-choice question (MCQ) frameworks used to assess human interpretation (Lee et al., 2016; Cui et al., 2023; Ge et al., 2023).

We collaborated with data visualization experts to develop a comprehensive misleader taxonomy (fig. 2), covering 60 unique (misleader, chart type) pairs across 21 misleaders and 10 chart types (fig. 7). For each pair, experts authored 2–3 well-defined examples, resulting in a total of 155 seed MCQs, which were standardized into D3.js (Bostock et al., 2011) visualizations, CSV data, and labeled JSON formats. Using automated expansion and extensive expert review involving 20 trained reviewers, we constructed a high-quality dataset of 3,026 curated misleading chart MCQs. We benchmark 24 state-of-the-art MLLMs and conduct systematic analysis across misleader types, chart formats, and error patterns to assess their capabilities. To support future progress, we propose a Region-Aware Misleader Reasoning pipeline that enhances MLLM performance by explicitly guiding attention to misleading chart regions.

2 Misleading ChartQA Benchmark

In this section, we describe the construction of the Misleading ChartQA dataset, which involves four main stages: (1) Misleader Taxonomy Construction, (2) Seed MCQ Design, (3) MCQ Augmentation and Iterative Refinement, and (4) Intensive Expert Validation.

2.1 Misleader Taxonomy Construction

To capture the diverse ways visualizations can mislead, we constructed a Misleader Taxonomy by consolidating deceptive strategies from academic literature and three publicly available collections of real-world misleading visualizations (Lo et al., 2022; Börner et al., 2019; Lan and Liu, 2024). Four data visualization experts—two postdoctoral researchers and two senior PhD students—independently reviewed these sources to compile an initial list of common misleaders. Through collaborative refinement, they merged overlapping items, clarified ambiguous definitions, and removed overly narrow cases, resulting in 21 distinct misleader types. The experts then mapped relevant chart types to each misleader, focusing on contexts where these deceptive patterns frequently occur. This process yielded 10 unique chart types and 60 distinct (misleader, chart type) pairings, ensuring broad and representative coverage. Detailed definitions and chart mappings are provided in fig. 7. Finally, the misleaders were organized into a structured taxonomy (fig. 2), forming the foundation for subsequent data augmentation.

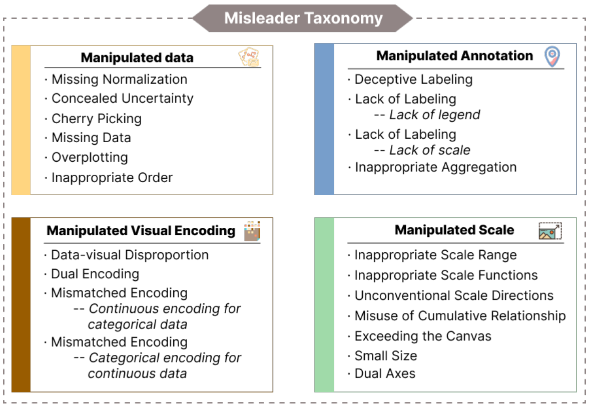


Figure 2: The taxonomy categorizes 21 misleaders into four groups based on manipulation techniques.

2.2 Seed Multiple-Choice Question Design

Building on our Misleader Taxonomy and the 60 (misleader, chart type) pairs, we collaborated with four experts to construct a comprehensive set of "seed MCQs", ensuring coverage of all pairings with multiple examples per pair. This seed set was derived from two primary sources. First, experts manually reviewed MCQs from prior studies (Lee et al., 2016; Cui et al., 2023; Ge et al., 2023), identifying those that aligned with our taxonomy and pairing scheme. An MCQ was selected if at least three out of four experts agreed it was a good match for a specific (misleader, chart type) pair. This process yielded 122 MCQs covering 49 of the 60 pairs.

For the remaining 11 uncovered pairs, each expert independently crafted new misleading chart QA items, which were then refined and finalized through multiple rounds of collaborative discussion. This led to an additional 33 MCQs. In total, we compiled 155 seed MCQs, ensuring that each (misleader, chart type) pairing is represented by 2–3 well-defined examples.

As shown in fig. 1, each seed MCQ includes: (1) a misleading chart, (2) a corresponding question, (3) multiple answer choices, (4) labeled correct and misleading answers, and (5) metadata with an explanation of the misleading aspect. Once finalized, all seed MCQs were encoded in a standardized format to support systematic chart and data variation. Each encoded MCQ consists of:

Misleading Chart Code Implementation. To enable flexible generation and variation of misleading chart visualizations, each seed chart was implemented using D3.js (Bostock et al., 2011), a JavaScript library for highly customizable visualizations. The code was structured in modular HTML files for easy rendering, consistent coding style, and efficient generation of visual variations.

CSV Data and JSON QA Specification. Each chart was paired with a curated CSV dataset designed to reflect the associated misleader scenario. For instance, a scatter plot labeled as *Cherry Picking* may use a selectively filtered dataset to exaggerate a trend (e.g., appendix A.9). Corresponding MCQs were encoded in JSON format, including question text, answer choices, correct and misleading answers, and detailed metadata for compatibility and downstream processing.

Chart Figure Generation. We rendered each chart using the implemented code and data, and developed a labeling tool (fig. 8) for experts to annotate misleading regions using bounding boxes. Both raw and annotated chart images were exported in standardized JPEG format with consistent dimensions to support scalable dataset expansion.

2.3 MCQs Augmentation and Refinement

Using seed MCQs for each misleader–chart type pair, we conduct a data augmentation process, leveraging general world knowledge from MLLMs (e.g., GPT-4o) to generate diverse MCQ variations while preserving the core misleading features.

Specifically, we apply controlled perturbations to chart code and introduce randomized yet plausible variations to the CSV data. This process does not rely on the model's training data, proprietary knowledge, or internal mechanisms, but instead uses only its general reasoning ability. By design, it minimizes the risk of model bias or knowledge leakage, ensuring that augmented examples for later experiments reflect generic reasoning rather than model-specific heuristics. The next section outlines the workflow structure, with detailed prompt templates in appendix A.11.1.

For each seed question, the annotated chart image, code, data, and JSON QA specification serve as core inputs to our MLLM-powered augmentation pipeline. We use ChatGPT-4o for its strong performance and efficiency, while strictly limiting its role to general-purpose tasks such as modifying HTML object attributes (e.g., color, axis scale, label position) and introducing plausible random adjustments to CSV data. These actions rely solely on general world knowledge and do not require any model-specific internal training data. The augmentation process consists of two main stages—*Chart Variation* and *QA Generation*—followed by an *Automated Evaluation, Feedback, and Refinement Loop* to ensure high-quality outputs.

Chart Variation: In the first stage (fig. 3-A), we apply controlled modifications to the chart code and underlying dataset to generate visual and contextual diversity. Specifically, the MLLM perturbs the seed D3.js code by adjusting general HTML attributes such as color schemes, axis layout, font size, or chart titles—tasks based on common web development conventions. Simultaneously, the associated CSV data is modified through random perturbations of numeric values and category labels, while maintaining the overall distribution and preserving the intended misleading effect. This stage ensures that each variation preserves the original misleader but presents it in a new surface form suitable for robust model benchmarking.

QA Generation: Once the chart and dataset are modified, the pipeline (fig. 3-B) launches a local server to render the updated chart and capture it as

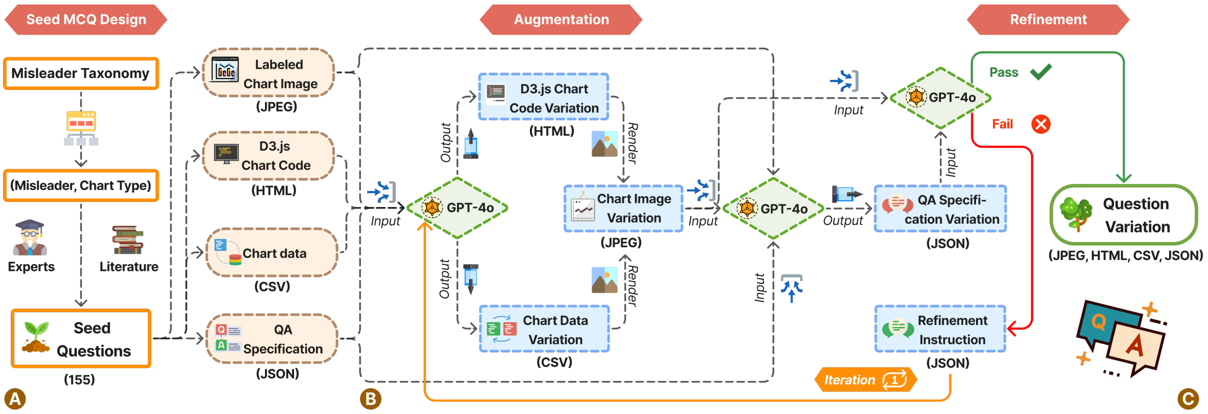


Figure 3: Overview of the Automated MCQ Augmentation and Iterative Refinement workflow. (A) *Seed MCQ Design:* Questions are authored by experts, guided by the proposed misleader taxonomy and relevant literature. (B) *Chart Variation:* MLLM modifies chart code and data to generate variations while preserving the intended misleader. (C) *MCQ Augmentation and Refinement Loop*. A separate MLLM generates QA pairs and explanations, followed by an evaluation and revision loop to improve failed cases. Final outputs include variations in JPEG, HTML, CSV, and JSON.

an image. This image, along with the original seed QA specification and metadata, is then passed to another MLLM module, which adjusts the MCQ to align with the new chart while preserving the original misleading logic.

Automated Evaluation, Feedback, and Refinement Loop: To ensure quality and reduce manual effort in the final review stage, each generated QA pair undergoes an automated, iterative first-pass check and revision process using an MLLM module. This module assesses whether the question, chart, and answers are logically coherent and whether the intended misleader is accurately preserved. If issues such as erroneous charts, ambiguous questions, or visual-question mismatches are detected, the system provides targeted revision instructions. These revisions are fed back into the generation module in a loop that continues until the output passes all checks. By filtering and correcting obvious errors early, this process significantly reduces the burden on human reviewers. At the end of this automated stage, a total of 4,263 augmented QA samples were generated across all misleader–chart type combinations, ready for subsequent expert validation.

2.4 Intensive Expert Validation

While automation filters low-quality outputs, expert validation remains crucial to ensure each augmented MCQ meets high standards. Due to the nuance of misleading charts, this stage requires intensive expert effort and cannot be reliably delegated to crowd-sourced or general annotators.

To this end, we recruited 20 PhD students specializing in data visualization—individuals with deep expertise in chart design, cognitive perception, and visual literacy—specifically to handle the complex reasoning required to evaluate misleading visual content. Each expert was compensated at $30 USD per hour and followed a three-stage evaluation process using our custom annotation tool (fig. 8). This process involved verifying whether the chart reflects the intended misleader, assessing the clarity and validity of the chart and QA pair, and deciding whether to reject, revise, or approve each sample (appendix A.5).

Across the 4,263 augmented QA samples, 29.02% were filtered out due to misalignment or irreparable chart issues, 60.52% were revised by updating the QA content, explanation, or making simple adjustments to the chart code, and 10.46% approved without modification. Each approved sample was reviewed by at least two experts, and revised samples underwent an additional validation round. This layered process ensured that all retained samples met strict standards. The final dataset contains 3,026 expert-validated MCQs, with corresponding charts, data, QA specifications, and misleader annotations. A detailed dataset breakdown and benchmark comparison are provided in table 3.

3 Experiments

In this section, we first describe our experimental setup (section 3.1), followed by a comprehensive evaluation results on the Misleading ChartQA

Model	Baseline			Zero-shot CoT			Pipeline		
	W. O.	W. M.	Acc.	W. O.	W. M.	Acc.	W. O.	W. M.	Acc.
Random Guess	50.00	25.00	25.00	50.00	25.00	25.00	50.00	25.00	25.00
Average (Overall)	27.38	35.02	37.60	28.35	34.51	37.14	26.82	33.43	39.76
Closed-source									
GPT-4o	26.60	38.47	**34.93**	25.57	37.79	**36.64**	27.74	33.22	39.04
GPT-4.1	21.92	43.15	34.93	19.86	44.29	35.84	22.60	37.21	40.18
GPT-o1	30.02	35.62	**34.36**	24.43	37.44	**38.13**	23.29	34.02	**42.69**
GPT-o3	23.29	39.95	**36.76**	26.94	39.95	**33.11**	23.06	34.93	42.01
GPT-o4-mini	22.60	39.95	**37.44**	24.43	39.95	**35.62**	25.11	36.07	38.81
Claude-3.5-Sonnet	36.30	29.57	34.13	27.63	35.38	36.99	25.80	35.96	38.24
Claude-3.7-Sonnet	35.16	30.59	34.25	27.63	34.59	37.78	37.21	37.78	25.01
Gemini-2.0-Flash	43.49	25.46	31.05	47.03	18.04	34.93	42.58	20.78	36.64
Gemini-2.5-Flash	43.15	18.95	**37.90**	39.50	20.09	**40.41**	37.44	25.11	37.44
Average (Closed-Source)	31.39	33.52	35.08	29.22	34.17	36.61	29.43	32.79	37.78
Open-source									
DeepSeek-VL2-Tiny	28.54	40.52	**30.94**	32.88	37.90	**29.22**	31.74	35.27	32.99
DeepSeek-VL2-Small	26.60	43.61	**29.79**	34.70	44.06	**21.24**	27.40	43.15	29.45
DeepSeek-VL2	26.48	43.61	29.91	30.37	34.70	34.93	24.43	38.58	36.99
Qwen2.5-VL-3B	35.16	30.60	**34.24**	36.99	29.22	**33.79**	34.70	27.63	37.67
Qwen2.5-VL-7B	27.40	34.93	37.67	29.22	33.11	37.67	27.63	31.74	40.64
Qwen2.5-VL-72B	29.45	29.45	**41.10**	28.77	28.77	**42.47**	31.51	25.11	43.38
InternVL2.5-4B-MPO	24.20	39.73	36.07	28.77	33.33	37.90	26.48	36.07	37.44
InternVL2.5-8B-MPO	19.86	38.36	41.78	22.61	34.70	42.69	18.72	36.53	44.75
InternVL2.5-26B-MPO	20.78	36.76	42.47	29.22	29.68	41.10	18.49	38.81	42.69
InternVL2.5-78B-MPO	20.09	31.96	47.95	16.89	36.76	46.35	18.95	32.31	48.74
InternVL3-8B-MPO	26.48	31.51	42.01	33.56	37.79	28.65	25.57	30.59	43.84
InternVL3-38B-MPO	17.81	34.47	47.72	19.18	39.50	41.32	20.78	35.16	44.06
InternVL3-78B-MPO	16.89	33.11	**50.00**	17.48	32.19	**50.23**	18.72	29.34	**51.94**
Average (Open-Source)	24.60	36.05	39.36	27.74	34.75	37.50	25.01	33.87	41.12

Table 1: Overall evaluation results of different MLLMs on Misleading ChartQA across three methods: Baseline, zero-shot CoT, and our proposed Pipeline (section 3.3). ***W.O.*** refers to errors from general distractors, ***W.M.*** from the misleading distractor, and ***Acc.*** denotes accuracy (selection of the correct answer). Prompt templates are detailed in appendices A.11.2 and A.11.3.

benchmark (section 3.2). Full implementation details are provided in the appendix A.6.

3.1 Experimental Setup

To comprehensively evaluate model performance on the Misleading ChartQA benchmark, we cover most recent widely used MLLMs, spanning both closed-source `GPT` series (4o, 4.1, o1, o3, o4-mini) (OpenAI, 2024a,b), `Claude` series (3.5 & 3.7 Sonnet) (Anthropic, 2024, 2025), and `Gemini` series (2.0 & 2.5 Flash) (Deepmind, 2024, 2025), as well as open-sourced `DeepSeek-VL2` (Wu et al., 2024b), `Qwen2.5-VL` (Bai et al., 2025), and `InternVL2.5 & InternVL3` (Chen et al., 2024b), with parameter sizes ranging from 2B to 78B.

For each model, we adopt the default prompting configurations from their respective papers or official documentation as the baseline (Chen et al., 2024b; DeepLearning.AI, 2025). We additionally apply the zero-shot Chain-of-Thought (CoT) prompting strategy (Kim et al., 2023) to examine how prompting affects performance on misleading questions. Finally, we compare both settings with our proposed Region-Aware Misleader Reasoning approach (referred to as *Pipeline*, detailed in section 3.3) to demonstrate its effectiveness.

3.2 Main Results

The overall results are presented in table 1, from which we can make the following observations:

(1) **The Misleading ChartQA task is highly challenging**, with most models scoring around 40% and the best-performing model reaching only 50.00% accuracy. This contrasts sharply with other chart-related benchmarks, where state-of-the-art models typically score around 90%. Notably, prior research similar performance from the general public on misleading chart comprehension tests, averaging 39% (SD = 16%) (Ge et al., 2023). These findings suggest that current MLLMs, trained pri-

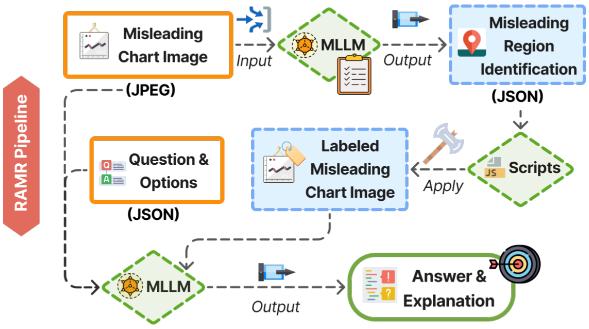


Figure 4: The Region-Aware Misleader Reasoning (RAMR) pipeline guides MLLMs to localize misleading regions first and generate answers using both original and labeled chart inputs.

marily on general corpora, perform comparably to humans and lack sufficient exposure to misleading charts—underscoring the need for a dedicated corpus and further research on this task.

(2) **MLLMs Are More Likely to Be Misled Than Distracted by Regular Distractors.** Across all settings, MLLMs are more prone to selecting misleading distractors (*W.M.*) than generic ones (*W.O.*), despite the 2:1 ratio favoring *W.O.* in random guessing. Under the baseline, *W.M.* averages 36.05% (open-source) and 33.52% (closed-source), notably exceeding the *W.O.* rates of 24.60% and 31.39%, respectively. This pattern persists across CoT and *Pipeline* settings. Even the lowest *W.M.* (32.78% in closed-source *Pipeline*) remains high. These results suggest MLLMs can ignore irrelevant options but still struggle to recognize and reason through deceptive chart cues, revealing a core weakness in visual critical reasoning.

(3) **Open-Source MLLMs Surpass Closed-Source Models on Misleading Charts.** Open-source models consistently surpass closed-source ones across all settings. In the baseline, they average 39.36% accuracy versus 35.08% for closed-source models—a trend that holds under both CoT and Pipeline settings. Most notably, InternVL3-78B-MPO achieves the highest scores across all settings: 50.00% (Baseline), 50.23% (CoT), and 51.94% (Pipeline), significantly outperforming all closed-source models (with o1 & Gemini-2.5 as the top performers). These results underscore the growing strength of open-source MLLMs in nuanced visual reasoning under large-scale parameters.

(4) **Impact of Chain of Thought (CoT) Reasoning.** To align with prior benchmarks (Kim et al., 2023; DeepLearning.AI, 2025; Chen et al., 2024b), we adopt a zero-shot CoT setting. It yields gains for most closed-source models (e.g., GPT-4o: 34.93% → 36.64%, Gemini-2.5-Flash: 37.90% → 40.41%), except for o3 and o4-mini—likely due to their already strong inherent reasoning abilities. In contrast, open-source models show limited or even negative effects: small and mid-sized models (e.g., DeepSeek-VL2-Tiny/Small, Qwen2.5-VL-3B) exhibit performance drops, while larger models (e.g., InternVL3-78B, Qwen2.5-VL-72B) gain only 0.5-1%. These results indicate that while CoT brings modest gains in some cases, it remains insufficient for handling misleading visual elements—especially in open-source models—highlighting the need for strategies that explicitly guide attention to deceptive features.

3.3 Region-Aware Misleader Reasoning

To enhance MLLMs' performance on Misleading ChartQA, we propose a multi-stage pipeline called Region-Aware Misleader Reasoning, inspired by how domain experts examine deceptive visualizations. This approach first identifies deceptive chart elements only, incorporating external scripts to assist this step-by-step process.

As illustrated in fig. 4, the pipeline begins with an MLLM independently analyzing the chart using a misleader checklist and outputting a JSON file with the coordinates and explanations of suspected misleading regions. This output is then passed to a JavaScript script that overlays bounding boxes onto the original chart. In the second stage, both the labeled chart (with explanations) and the original chart, along with the question and options, are provided to another MLLM to generate the final answer. We include both chart versions improves robustness against mislabeling by treating the labeled chart as a reference rather than definitive ground truth.

As shown in table 1-*Pipeline* and discussed in section 3.2, our method consistently outperforms both baseline and zero-shot CoT settings across model families. Notably, it boosts the best closed-source model (GPT-o1) to 42.69% and the best open-source model (InternVL3-78B-MPO) to 51.94%. Prompt templates are detailed in appendices A.11.2 and A.11.3.

4 Discussion & Error Analysis

To better understand the limitations of current MLLMs on the Misleading ChartQA benchmark, we provide a diagnostic analysis of performance

Misleader		Wrong due to Others	Wrong due to Misleader	Accuracy
Manipulated Data	Cherry Picking	16.12	50.89	32.99
	Missing Data	33.86	35.31	30.83
	Overplotting	33.18	40.24	26.58
	Inappropriate Order	32.71	35.63	31.66
	Missing Normalization	27.01	44.27	28.72
	Concealed Uncertainty	30.96	37.40	31.64
	Category Overall	**28.97**	**40.62**	**30.40**
Manipulated Annotation	Deceptive Labeling	21.05	45.88	33.07
	Lack of Labeling $_{\text{Lack of legend}}$	34.66	33.08	32.26
	Lack of Labeling $_{\text{Lack of scales}}$	30.19	32.72	37.09
	Inappropriate Aggregation	34.78	9.06	56.16
	Category Overall	30.17	30.19	39.64
Manipulated Visual Encoding	Dual Encoding	33.00	27.61	39.39
	Data-visual Disproportion	39.28	18.99	41.73
	Mismatched Encoding $_{\text{Continuous encoding}}$	28.32	32.63	39.05
	Mismatched Encoding $_{\text{Categorical encoding}}$	28.66	27.17	44.17
	Category Overall	**32.31**	**26.60**	**41.09**
Manipulated Scale	Small Size	37.17	23.14	39.69
	Dual Axes	31.27	35.65	33.08
	Exceeding the Canvas	32.46	29.23	38.31
	Inappropriate Scale Range	37.62	33.05	29.33
	Inappropriate Scale Functions	28.58	27.29	44.13
	Unconventional Scale Directions	11.62	62.96	25.42
	Misuse of Cumulative Relationship	32.95	28.60	38.45
	Category Overall (normalized)	30.24	34.27	34.69

Table 2: Summary statistics for different misleader categories and types, showing average rates of *Wrong due to Others*, *Wrong due to Misleader*, and overall accuracy.

across misleader types and chart structures, followed by an error analysis of failure cases.

4.1 Performance Across Misleader Types

As shown in table 2, MLLMs perform worst on the **Manipulated Data** group, which records the lowest *Accuracy* (30.40%), the lowest *Wrong due to Others* rate (28.97%), whereas the highest *Wrong due to Misleader* rate (40.62%). This suggests that models are likely to be misled by subtle data distortions (e.g., *Cherry Picking*, *Missing Normalization*). In contrast, the **Manipulated Visual Encoding** group exhibits the highest average *Accuracy* (41.09%) and the lowest *Wrong due to Misleader* rate (26.60%), indicating that MLLMs are more proficient at detecting visually apparent issues such as *Dual Encoding* and *Mismatched Encoding*.

These findings highlight a key limitation in MLLMs' reasoning: they are more adept at spotting visual discrepancies than interpreting manipulations that affect the underlying data semantics. We hypothesize this stems from a pretraining bias—models are often optimized for aligning text with visible elements rather than performing deeper statistical inference and understanding. Example MCQs from these two categories are provided in appendices A.9 and A.10

4.2 Performance Across Chart Types

As shown in fig. 5, MLLMs exhibit varied performance across different chart types. **Line Charts** achieve the highest accuracy (39.44%), followed by **Area Charts** (39.21%), **Pie Charts** (34.64%), suggesting relatively strong model performance on conventional chart formats. Conversely, formats such as **Choropleth Maps** (26.97%) and **Stacked Area Charts** (28.26%) show the lowest accuracies, indicating persistent challenges in interpreting spatial or layered visual structures accurately.

However, digging into the error types reveals that simpler chart types appear more susceptible to misleading cues than complex ones. **Bar Charts** (37.85%), **Line Charts** (37.28%), and **Area Charts** (35.05%) all exhibit high *Wrong due to Misleader* rates, while maintaining relatively low *Wrong due to Others* rates (26.75%, 23.28%, and 25.74%). This suggests that visual simplicity may make it easier to apply subtle deceptive manipulations, leading models to overlook them—mirroring patterns observed in human reasoning.

In contrast, complex chart types show the opposite trend. The **Stacked Area Chart** has the lowest misleader error rate (13.77%) but the highest *Wrong due to Others* rate (57.97%), suggesting reasoning breakdowns even without deceptive

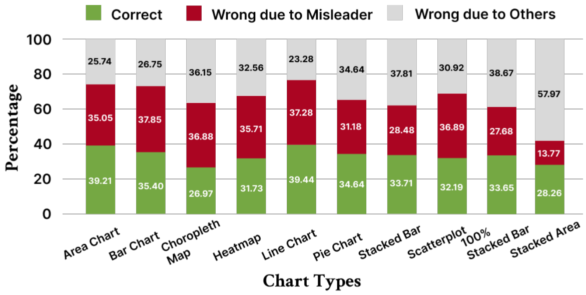


Figure 5: MLLM performance by chart type, revealing weak reasoning on complex stacked charts and high misleader susceptibility in simpler ones.

cues. **Stacked Bar Charts** and 100% **Stacked Bar Charts** follow a similar pattern, with low misleader errors (28.48%, 27.68%) but high regular distractor rates (37.81%, 38.67%). **Choropleth Maps** also show low accuracy (26.97%) and high general error (36.15%).

These results indicate that current MLLMs struggle with the structural reasoning required by complex layouts such as stacked series and geographic maps—even in the absence of explicit misleaders. A likely explanation is limited exposure during pretraining, as these formats are underrepresented in existing benchmarks such as ChartQA (Masry et al., 2022), ChartLlama (Han et al., 2023), and ChartInsights (Wu et al., 2024a). Our inclusion of diverse structured chart types (e.g., stacked and geographic charts) thus adds critical diagnostic value for evaluating visual reasoning.

4.3 Error Analysis

To better understand model limitations, we analyze failure cases from the top-performing models: GPT-o1 and InternVL3-78B-MPO, under the proposed pipeline. Three major error types emerge:

Misleading Region Localization Errors ($\approx$ 70%**).** The majority of failures stem from incorrect localization of misleading regions, leading to flawed downstream reasoning. Future research should focus on improving both the model's ability to identify misleading elements and its precision in generating accurate region coordinates.

Misleader Interpretation and Reasoning Errors. In some cases, the model correctly identifies the misleading region but fails to reason through its implications—such as recognizing a manipulated data order but not mentally reordering the data to recover the true trend. This suggests that accurate answer selection often requires not just detection of the misleader, but also corrective reasoning to reconstruct the intended information.

Question Misunderstanding. A smaller subset of errors arises from misinterpreting question intent, especially involving subtle qualifiers or conditional logic—such as confusing when to choose "Cannot be determined" versus directly answering "No". This suggests future work should go beyond evaluating option selection and include more fine-grained annotation of model reasoning, particularly in tasks like Misleading ChartQA where interpretive reasoning is central.

5 Related Works

Here we summarize key related work below and provide full details in appendix A.1.

Chart Reasoning Benchmarks. Prior benchmarks like ChartQA (Masry et al., 2022) and PlotQA (Methani et al., 2020) evaluate basic chart understanding on common chart types. Recent works expand chart coverage (Han et al., 2023; Xia et al., 2024), add task complexity (e.g., captioning (Huang et al., 2023), summarization (Rahman et al., 2023)). However, none explicitly focus on misleading visualizations.

Misleading Visualization Studies. Human-centered evaluations (Lee et al., 2016; Ge et al., 2023) have identified common chart misleaders and assessed reasoning via MCQs, but their limited scale is inadequate for benchmarking MLLMs. Taxonomy-driven studies (Lo et al., 2022; Lan and Liu, 2024) emphasize design heuristics over standardized tests.

MLLMs and Misleading Charts. Recent efforts (Bendeck and Stasko, 2024; Tonglet et al., 2025) evaluate MLLMs on small sets of human-designed misleading charts, offering limited generalizability. In contrast, we propose the first large-scale benchmark and conduct a comprehensive evaluation of 24 MLLMs.

6 Conclusions

We present Misleading ChartQA, the first benchmark for evaluating MLLMs' ability to detect and reason about misleading chart visualizations. The dataset comprises over 3,000 curated examples across 21 misleader types and 10 chart formats. We benchmark 24 MLLMs, conduct systematic analyses, and introduce a pipeline to improves model accuracy. Our work lays a foundation for advancing MLLM-based visual misinformation detection and robust chart comprehension.

Limitations

Limited Visual Prompt Design and Comparison In line with the original models publishers' approaches (e.g., Qwen, DeepSeek, and InternVL series), which primarily use zero-shot methods for ChartQA benchmark testing, our evaluation also adopts a zero-shot approach. While this alignment facilitates comparison, it is likely that MLLMs' performance could be further enhanced through few-shot learning methods. Future work could explore this by incorporating few-shot techniques to potentially improve the models' capabilities in handling misleading chart detection tasks.

Lack of Fine-Tuning on MLLMs We did not explore fine-tuning methods to improve MLLMs' performance on this task. The main reason for this is our goal of first obtaining a comprehensive understanding of the performance of the latest generation of MLLMs on Misleading Chart QA. Based on the results of our experiments, future research could investigate fine-tuning, particularly with the InternVL2-5-78B-MPO model, which exhibited the strongest performance among all the models tested.

References

Anthropic. 2024. Claude 3.5 Sonnet Model Card Addendum.

Anthropic. 2025. Claude 3.7 Sonnet.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Alexander Bendeck and John Stasko. 2024. An empirical evaluation of the gpt-4 multimodal language model on visualization literacy tasks. *IEEE Transactions on Visualization and Computer Graphics*.

Katy Börner, Andreas Bueckle, and Michael Ginda. 2019. Data visualization literacy: Definitions, conceptual frameworks, exercises, and assessments. *Proceedings of the National Academy of Sciences*, 116(6):1857–1864.

Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. 2011. D^3 data-driven documents. *IEEE transactions on visualization and computer graphics*, 17(12):2301–2309.

Jeremy Boy, Ronald A Rensink, Enrico Bertini, and Jean-Daniel Fekete. 2014. A principled way of assessing visualization literacy. *IEEE transactions on visualization and computer graphics*, 20(12):1963–1972.

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

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, and 1 others. 2024b. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

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

Yuan Cui, W Ge Lily, Yiren Ding, Fumeng Yang, Lane Harrison, and Matthew Kay. 2023. Adaptive assessment of visualization literacy. *IEEE Transactions on Visualization and Computer Graphics*.

DeepLearning.AI. 2025. ChatGPT Prompt Engineering for Developers - DeepLearning.AI.

Deepmind. 2024. Gemini 2.0 Flash.

Deepmind. 2025. Gemini 2.5 Flash.

Lily W Ge, Yuan Cui, and Matthew Kay. 2023. Calvi: Critical thinking assessment for literacy in visualizations. In *Proceedings of the 2023 CHI conference on human factors in computing systems*, pages 1–18.

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*.

Jiayi Hong, Christian Seto, Arlen Fan, and Ross Maciejewski. 2025. Do llms have visualization literacy? an evaluation on modified visualizations to test generalization in data interpretation. *IEEE Transactions on Visualization and Computer Graphics*.

Kung-Hsiang Huang, Mingyang Zhou, Hou Pong Chan, Yi R Fung, Zhenhailong Wang, Lingyu Zhang, Shih-Fu Chang, and Heng Ji. 2023. Do lvlms understand charts? analyzing and correcting factual errors in chart captioning. *arXiv preprint arXiv:2312.10160*.

Darrell Huff. 2023. *How to lie with statistics*. Penguin UK.

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

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

Melita Kennedy, Steve Kopp, and 1 others. 2000. *Understanding map projections*, volume 8. Esri Redlands, CA.

Seungone Kim, Se June Joo, Doyoung Kim, Joel Jang, Seonghyeon Ye, Jamin Shin, and Minjoon Seo. 2023. The cot collection: Improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning. *arXiv preprint arXiv:2305.14045*.

Gary King. 1986. How not to lie with statistics: Avoiding common mistakes in quantitative political science. *American Journal of Political Science*, pages 666–687.

Xingyu Lan and Yu Liu. 2024. “i came across a junk”: Understanding design flaws of data visualization from the public’s perspective. *IEEE Transactions on Visualization and Computer Graphics*.

Sukwon Lee, Sung-Hee Kim, and Bum Chul Kwon. 2016. Vlat: Development of a visualization literacy assessment test. *IEEE transactions on visualization and computer graphics*, 23(1):551–560.

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*.

Leo Yu-Ho Lo, Ayush Gupta, Kento Shigyo, Aoyu Wu, Enrico Bertini, and Huamin Qu. 2022. Misinformed by visualization: What do we learn from misinformative visualizations? In *Computer Graphics Forum*, volume 41, pages 515–525. Wiley Online Library.

Leo Yu-Ho Lo and Huamin Qu. 2024. How good (or bad) are llms at detecting misleading visualizations? *IEEE Transactions on Visualization and Computer Graphics*.

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*.

Morgan McFall-Johnsen. 2020. A ’cuckoo’ graph with no sense of time or place shows how Georgia bungled coronavirus data as it reopens.

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

Lotti O’Brien. 2024. Maps of world ’completely misleading’ as true size of Europe, China and Africa revealed.

OpenAI. 2024a. Hello GPT-4o.

OpenAI. 2024b. Introducing OpenAI o1.

Raian Rahman, Rizvi Hasan, Abdullah Al Farhad, Md Tahmid Rahman Laskar, Md Hamjajul Ashmafee, and Abu Raihan Mostofa Kamal. 2023. Chartsumm: A comprehensive benchmark for automatic chart summarization of long and short summaries. *arXiv preprint arXiv:2304.13620*.

Jonathan Tonglet, Tinne Tuytelaars, Marie-Francine Moens, and Iryna Gurevych. 2025. Protecting multimodal large language models against misleading visualizations. *arXiv preprint arXiv:2502.20503*.

Edward R Tufte and Peter R Graves-Morris. 1983. *The visual display of quantitative information*, volume 2. Graphics press Cheshire, CT.

Yifan Wu, Lutao Yan, Leixian Shen, Yunhai Wang, Nan Tang, and Yuyu Luo. 2024a. Chartinsights: Evaluating multimodal large language models for low-level chart question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 12174–12200.

Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, and 1 others. 2024b. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*.

Renqiu Xia, Bo Zhang, Hancheng Ye, Xiangchao Yan, Qi Liu, Hongbin Zhou, Zijun Chen, Min Dou, Botian Shi, Junchi Yan, and 1 others. 2024. Chartx & chartvlm: A versatile benchmark and foundation model for complicated chart reasoning. *arXiv preprint arXiv:2402.12185*.

Zhengzhuo Xu, Sinan Du, Yiyan Qi, Chengjin Xu, Chun Yuan, and Jian Guo. 2023. Chartbench: A benchmark for complex visual reasoning in charts. *arXiv preprint arXiv:2312.15915*.

Xingchen Zeng, Haichuan Lin, Yilin Ye, and Wei Zeng. 2024. Advancing multimodal large language models in chart question answering with visualization-referenced instruction tuning. *IEEE Transactions on Visualization and Computer Graphics*.

A Appendix

A.1 Full Related Works

A.1.1 Chart Reasoning Benchmarks

Chart Reasoning has emerged as a key area of focus within the vision-language community, with several benchmarks developed to assess models' abilities to interpret and reason about charts. Early datasets such as ChartQA(Masry et al., 2022) and PlotQA(Methani et al., 2020) primarily evaluated basic chart understanding, focusing on three common chart types. These datasets were relatively straightforward for recent MLLMs to solve. Subsequent benchmarks have either expanded chart type coverage (Han et al., 2023; Xia et al., 2024; Xu et al., 2023) or refined the complexity of tasks, distinguishing between high-level tasks (e.g., chart captioning, chart summarization (Kantharaj et al., 2022; Rahman et al., 2023; Cheng et al., 2023; Huang et al., 2023; Liu et al., 2022)) and low-level tasks (e.g., extracting numerical values (Kahou et al., 2017; Wu et al., 2024a)). Some works have also introduced more complex tasks such as chart structure extraction (Chen et al., 2024a). A detailed comparison of chart variety with existing benchmarks is provided in table 3 and fig. 9.

A.1.2 Misleading Chart Visualizations

Misleading chart visualizations have long been a significant topic in data visualization and human-computer interaction (King, 1986). Several standardized tests have been designed to evaluate human chart understanding and reasoning abilities (Lee et al., 2016; Boy et al., 2014; Börner et al., 2019). Recent efforts have evolved to emphasize critical thinking in chart comprehension, identifying around 10 categories of common misleaders in charts and formulating nuanced questions for human testing (Ge et al., 2023; Cui et al., 2023). However, these question sets consist of only about 40 questions, each addressing one or two examples of (misleader, chart type) combinations, which limits their effectiveness for evaluating MLLMs. Other latest studies have attempted to summarize common misleading visualization practices (Lo et al., 2022; Lan and Liu, 2024), but these focus on broad visualization design issues that do not directly apply to chart understanding tasks.

A.1.3 MLLMs in Misleading Chart Comprehension

Several recent studies have empirically evaluated MLLMs' performance in understanding misleading chart visualizations by testing them on existing standardized tests designed for humans (Bendeck and Stasko, 2024; Tonglet et al., 2025; Hong et al., 2025; Lo and Qu, 2024; Zeng et al., 2024). These studies typically involved a limited number of models and questions, making it difficult to draw reliable conclusions about MLLMs' ability. In contrast, our work constructs a diverse benchmark with over 3,000 samples, covering a broad range of misleaders and chart types. Through a comprehensive evaluation of 16 state-of-the-art MLLMs, we establish a strong foundation for this task first-ever.

A.2 Real-world examples: misleading charts

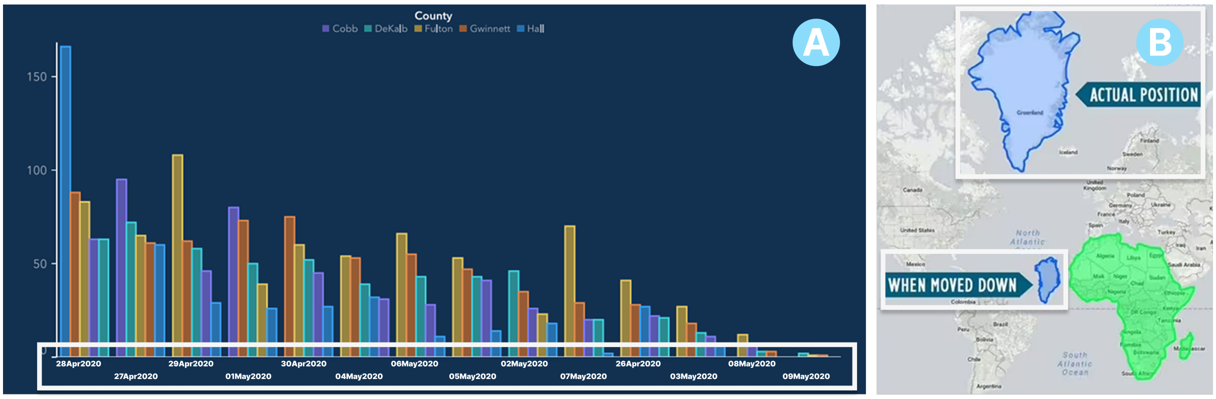


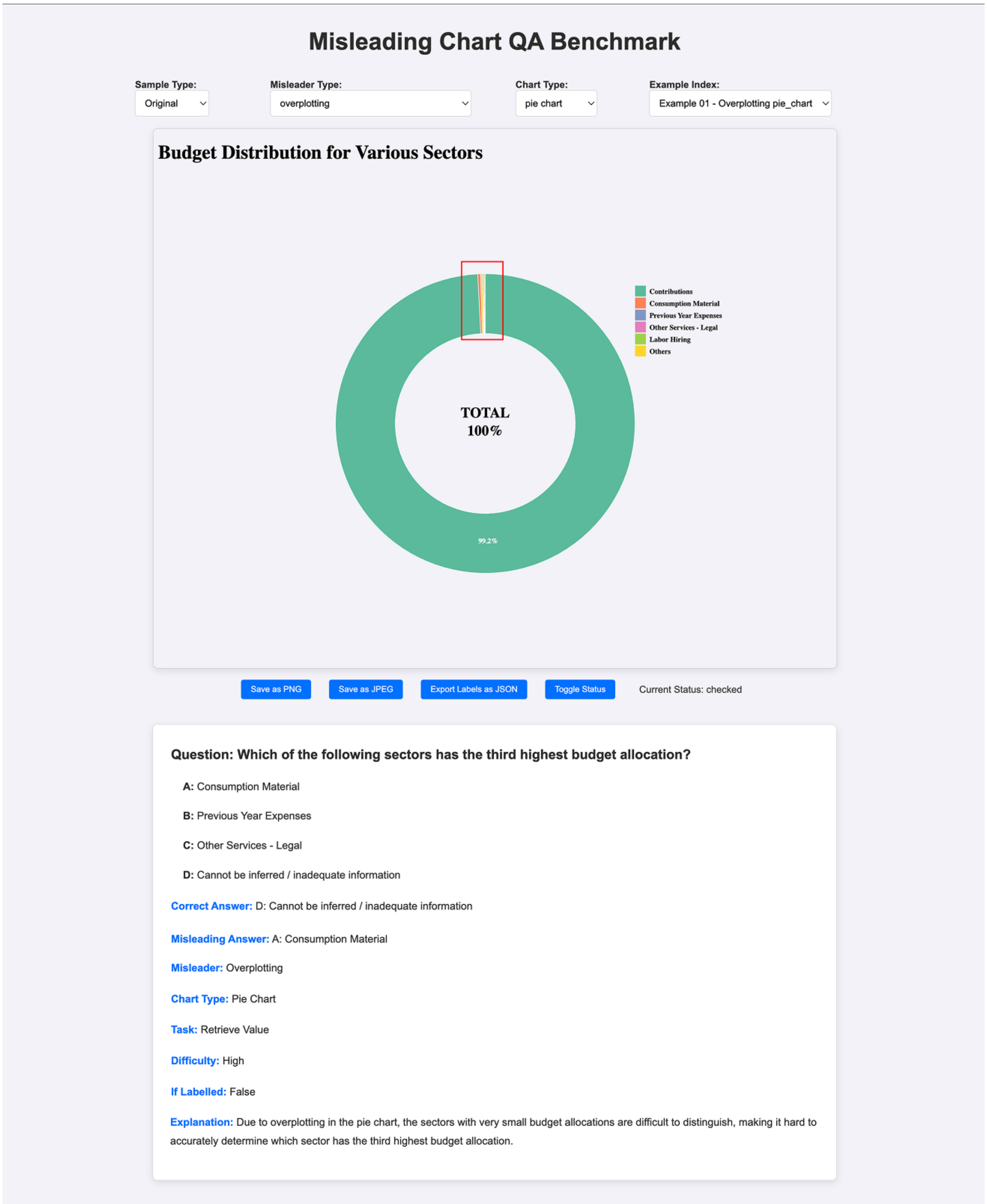
Figure 6: Two real-world examples of misleading chart visualizations. **(A)** A bar chart of COVID-19 cases across five counties, sorted by case count rather than by date, creating the false impression of a declining trend unless viewers carefully examine the x-axis. **(B)** The commonly used world map projection, which misrepresents Greenland as being the same size as Africa, despite Africa being significantly larger.

A.3 Misleader Definition

	Misleader Name	Definition	Area Chart	Bar Chart	Choropleth Map	Heatmap	Line Chart	Pie Chart	Stacked Area Chart	Stacked Bar Chart	Scatterplot	100% Stacked Bar Chart
Manipulated Data	**Missing Normalization**	Displaying unnormalized absolute values when relative or normalized comparisons would be more appropriate for interpretation.			✓							
	Concealed Uncertainty	Omitting uncertainty in visualizations can misrepresent the reliability of underlying data. In predictive contexts, this may lead viewers to develop an unjustified sense of confidence in the conclusions.		✓	✓						✓	
	Cherry Picking	Selecting only a subset of data to display, potentially misleading viewers by implying conclusions about the entire dataset.					✓				✓	
	Missing Data	Presenting a visual representation that suggests data exists when, in reality, it is missing.			✓							
	Overplotting	Overcrowding a visualization with excessive data points or elements, making it difficult to discern meaningful patterns.									✓	
	Inappropriate Order	Manipulating the order of data by manipulating axis labels or legend items in a way that misleads viewers or creates a false impression of trends.	✓	✓	✓						✓	✓
Manipulated Annotation	**Deceptive Labeling**	Using annotations or labels that contradict the data or make the visualization difficult to interpret.		✓			✓	✓				
	Lack of Labeling: *Lack of legend*	Omitting a legend that explains colors, symbols, or other encodings, leaving viewers uncertain about the meaning of the visualization.							✓	✓		
	Lack of Labeling: *Lack of scales*	Failing to provide axis scales or units of measurement, which can oversimplify or obscure the interpretation of the data.	✓	✓			✓					
	Inappropriate Aggregation	Combining or summarizing data in a way that distorts the true distribution or relationships, leading to inaccurate conclusions.	✓	✓			✓				✓	✓
Manipulated Visual Encoding	**Data-visual Disproportion**	Creating a visual representation where the graphical elements (e.g., bar heights) do not accurately correspond to the actual data values, leading to misinterpretation.		✓			✓	✓			✓	
	Dual Encoding	Using multiple visual channels (e.g., both width and height) to encode the same variable, which exaggerates the data's visual impact.		✓				✓				
	Mismatched Encoding: *Continuous encoding for categorical data*	Applying continuous encoding methods (e.g., color gradients or line connections) to categorical data, which can mislead viewers into perceiving relationships that do not exist.	✓				✓	✓				
	Mismatched Encoding: *Categorical encoding for continuous data*	Representing continuous data using discrete categories, potentially distorting trends and relationships.			✓	✓						
Manipulated Scale	**Inappropriate Scale Range**	Altering the scale of axes or legends by stretching, truncating, or using inconsistent binning, which distorts data representation.		✓	✓		✓		✓			✓
	Inappropriate Scale Functions	Applying arbitrary or misleading non-linear transformations to the scale of an axis, affecting how viewers perceive the relationships within the data.		✓			✓	✓				
	Unconventional Scale Directions	Using non-standard axis or legend orientations, such as inverting scales, which can confuse viewers and misrepresent relationships.	✓	✓	✓		✓				✓	
	Misuse of Cumulative Relationship	Incorrectly combining or accumulating data elements that do not logically sum or relate, distorting the true relationships.							✓	✓		✓
	Exceeding the Canvas	Allowing data points, labels, or visual elements to extend beyond the display area, causing loss of critical information.	✓	✓			✓					
	Small Size	Using excessively small text or graphical elements that hinder readability and make it difficult to interpret data.			✓		✓				✓	
	Dual Axes	Incorporating multiple axes in a way that complicates comparisons and forces viewers to mentally align different scales.					✓					

Figure 7: List of misleaders categorized under each misleader group, along with their detailed definitions and corresponding chart types. In total, there are 60 (misleader, chart type) pairings.

A.4 Expert Labeling Tool Interface



Misleading Chart QA Benchmark

Sample Type: Original

Misleader Type: overplotting

Chart Type: pie chart

Example Index: Example 01 - Overplotting pie_chart

Current Status: checked

Question: Which of the following sectors has the third highest budget allocation?

A: Consumption Material

B: Previous Year Expenses

C: Other Services - Legal

D: Cannot be inferred / inadequate information

Correct Answer: D: Cannot be inferred / inadequate information

Misleading Answer: A: Consumption Material

Misleader: Overplotting

Chart Type: Pie Chart

Task: Retrieve Value

Difficulty: High

If Labelled: False

Explanation: Due to overplotting in the pie chart, the sectors with very small budget allocations are difficult to distinguish, making it hard to accurately determine which sector has the third highest budget allocation.

Figure 8: Interface of our custom labeling tool used in the chart figure generation step. Experts annotate misleading regions using bounding boxes, as shown in the pie chart with an overplotting misleader. The interface also supports metadata editing, chart preview, and label export in standardized formats to facilitate expert validation and scalable dataset generation.

A.5 Expert Evaluation Guidelines

Overview

To ensure high-quality outputs in the *Misleading ChartQA* benchmark, each machine-generated MCQ was validated by PhD-level experts in data visualization. Experts used a custom labeling tool (Figure 8) to follow a structured 3-stage evaluation process guided by the protocol below.

Evaluation Protocol

Please review each sample (including the chart, question, answer options, and explanation) following the steps below:

1. **Verify Chart Correctness**
   - Does the chart clearly and accurately demonstrate the intended misleader?
   - Does it conform to the misleader definition in our taxonomy?
2. **Assess QA Pair Validity**
   - Does the question clearly and accurately reflect the misleading aspect?
   - Are the answer options logically sound?
   - Does the marked correct answer resolve the question as intended?
   - Does the marked misleading answer accurately reflect the misleading aspect as intended?
3. **Action Based on Assessment**
   - ***Reject:*** If the chart does not demonstrate the intended misleader, remove the sample.
   - ***Revise:*** If the chart is correct but the QA pair is problematic (e.g., vague question, incorrect or ambiguous answers), revise the QA pair accordingly.
   - ***Approve:*** If both the chart and QA pair are accurate and coherent, approve without modification.

Each approved sample was confirmed by at least two independent experts, and revised samples underwent an additional round of expert validation.

A.6 Implementation Details of Experiments

Our experiments were conducted on 8 NVIDIA A800 GPUs (80GB each) using PyTorch 2 and Python 3. Given the task's complexity, we selected only the most advanced versions of each model type and evaluated them across different parameter sizes. Due to computational constraints, we randomly sampled around 30% (876 cases) from the dataset, ensuring a balanced distribution across misleader and chart types for representativeness.

A.7 Comparison with related benchmarks

Task Focus	Datasets	#-Chart Types	# Chart	# Task type	Metadata?	Chart Code?	Chart Data?
Basic understanding	ChartQA	3	4.8k	4	N	N	N
	PlotQA	3	224k	1	N	N	N
Summarization/ captioning	ChartLlama	10	11k	7	N	N	N
	ChartBench	11	2.1k	4	N	N	N
	Chart-to-text	6	44k	3	N	N	N
	Chartsumm	3	84k	1	Y	N	N
Data/structure extraction	ChartInsights	7	2k	10	Y	N	N
	FigureQA	5	120K	6	N	N	N
Misleading Chart Comprehension	**Misleading ChartQA**	**10**	**3k**	**21**	**Y**	**Y**	**Y**

Table 3: Comparison of the Misleading ChartQA dataset with existing benchmarks. Misleading ChartQA is the first dataset specifically designed for the misleading chart comprehension task. It also features a diverse range of chart types and task types, along with rich metadata, chart code, and chart data.

A.8 Chart Types Distribution

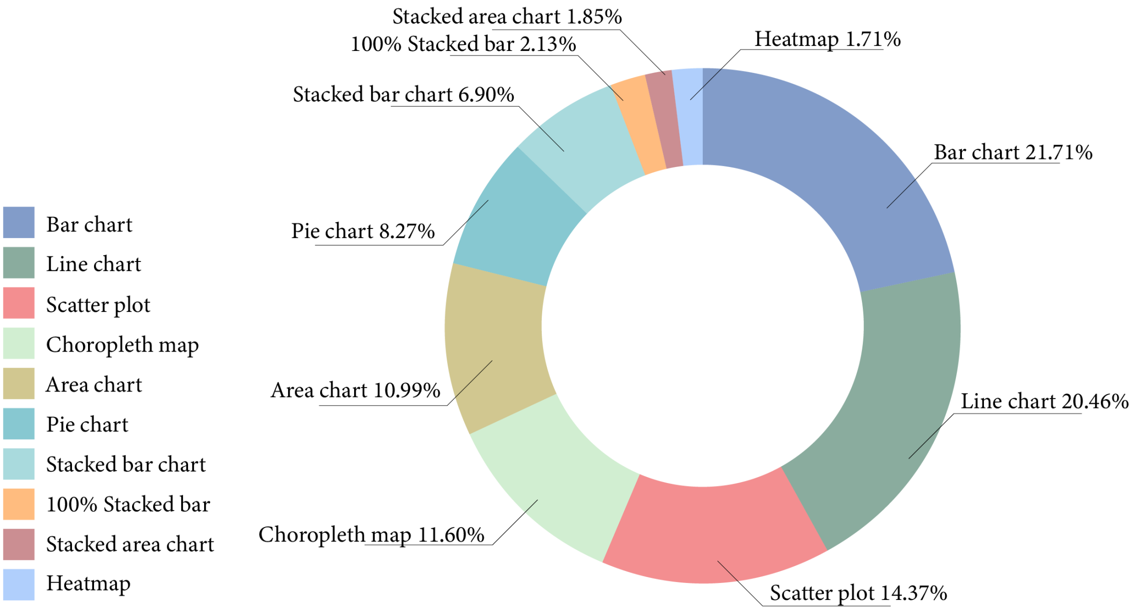
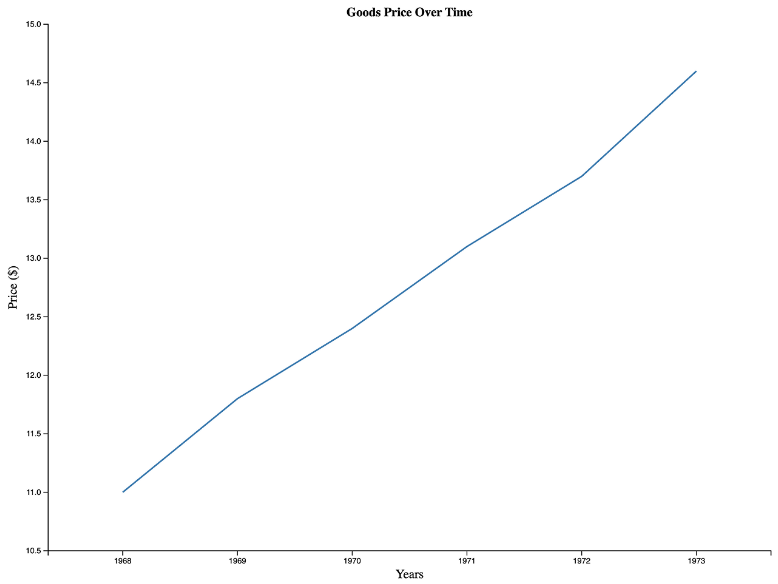


Figure 9: Breakdown of Chart Types in the Misleading ChartQA Dataset. **We intentionally balanced samples per (misleader, chart type) pair to reflect the natural mapping between chart types and supported misleaders** (e.g., heatmaps support fewer misleaders than bar charts). As a result, the overall chart distribution is uneven—mirroring real-world usage, where chart types like 100% stacked bars and stacked area charts are less common than bar or line charts.

A.9 Example: Manipulated Data

Goods Price Visualization



What is the trend of the price changes from 1968 to 1980 based on the provided chart data?

A: Generally increasing.

B: Generally decreasing.

C: Generally stable.

D: Cannot be inferred.

Correct Answer: D

Misleading Answer: A

Misleader: *Cherry Picking*

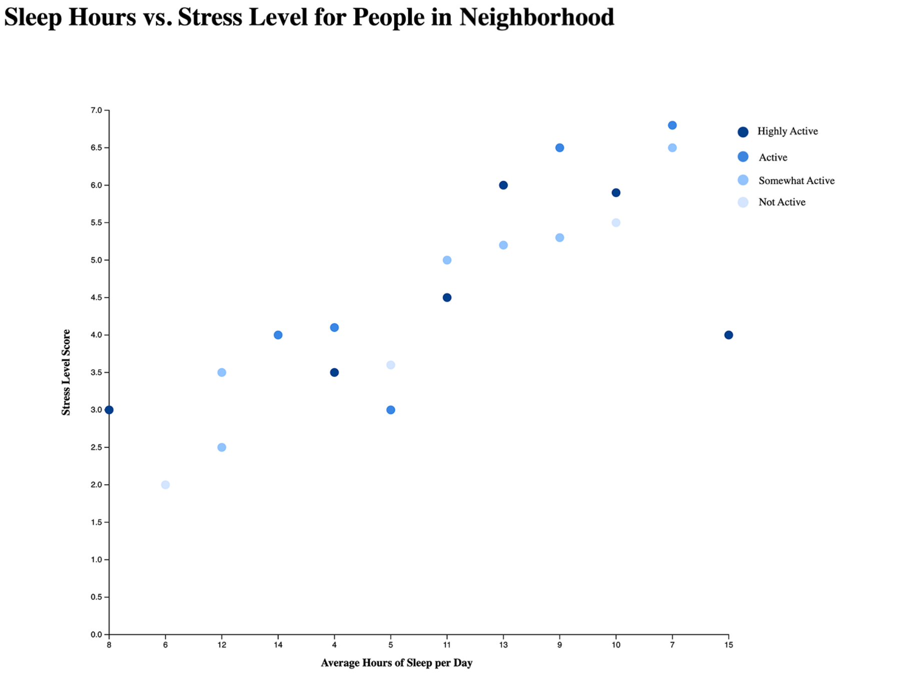
Chart Type: *Line Chart*

Task: *Correlations/Trends*

Difficulty: *Medium*

Explanation: *The chart shows an increasing trend during the selected period (1968-1973). However, when considering the broader data set (1973-1980), this snippet actually excludes important periods of decline, leading to a misleading interpretation.*

Figure 10: An example question from the **Manipulated Data** group, categorized under *Cherry Picking* and presented as a *Line Chart*.



Which of the following interpretations is most accurate based on the chart?

A: Individuals sleeping 9–10 hours tend to exhibit relatively high 'Stress Levels'.

B: People who sleep 4–5 hours are predominantly 'Not Active'.

C: Longer sleep duration always leads to higher 'Stress Level'.

D: None of the above.

Correct Answer: A

Misleading Answer: C

Misleader: *Inappropriate Order*

Chart Type: *Scatterplot*

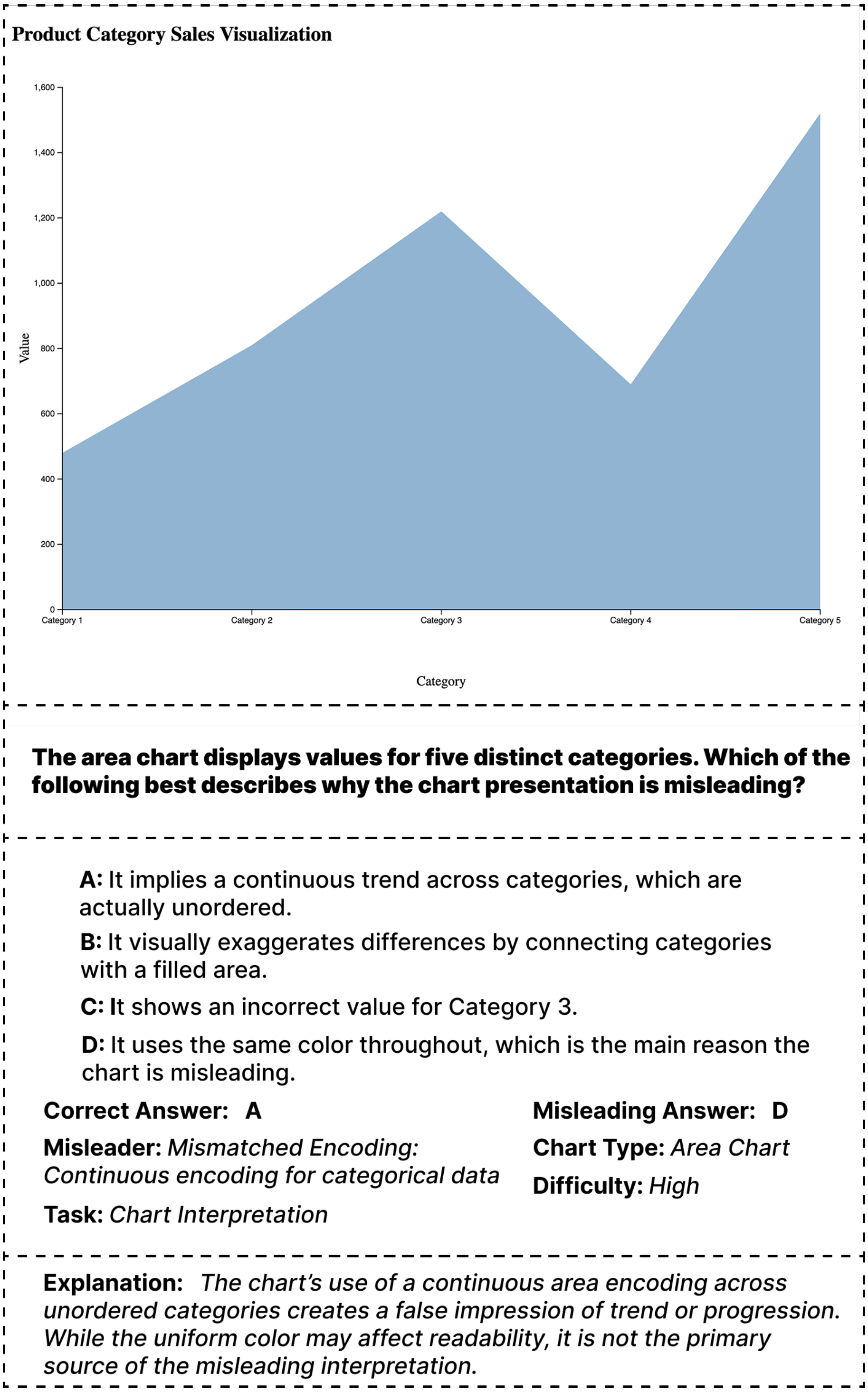
Task: *Make Comparisons Find Correlations/Trends*

Difficulty: *Medium*

Explanation: *Although a rough upward trend exists between sleep hours and stress level, the x-axis values are not arranged in numerical order. This inappropriate ordering can distort perceived patterns and lead viewers to misinterpret the relationship between variables.*

Figure 11: An example question from the **Manipulated Data** group, categorized under *Inappropriate Order* and presented as a *Scatterplot*.

A.10 Example: Manipulated Visual Encoding



The area chart displays values for five distinct categories. Which of the following best describes why the chart presentation is misleading?

A: It implies a continuous trend across categories, which are actually unordered.

B: It visually exaggerates differences by connecting categories with a filled area.

C: It shows an incorrect value for Category 3.

D: It uses the same color throughout, which is the main reason the chart is misleading.

Correct Answer: A

Misleader: *Mismatched Encoding: Continuous encoding for categorical data*

Task: *Chart Interpretation*

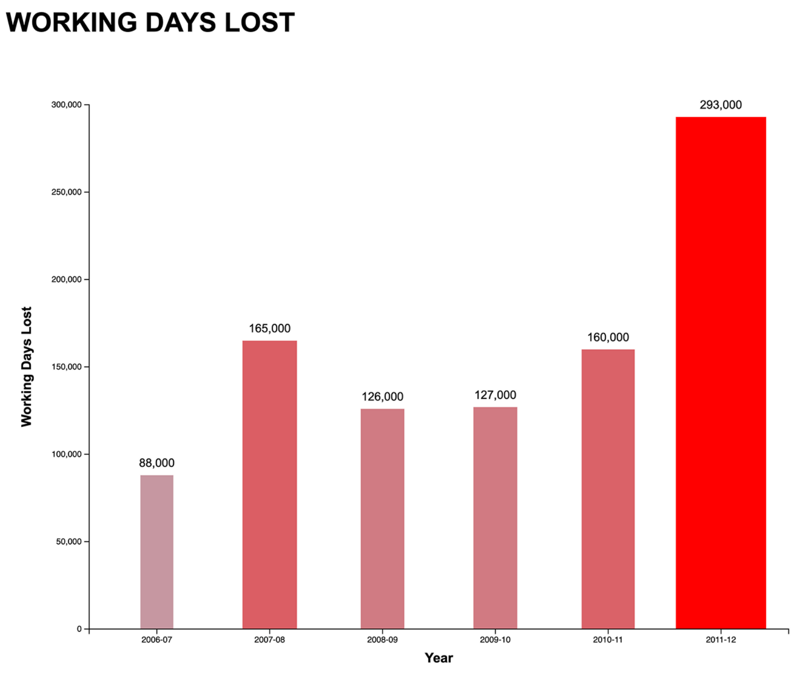
Misleading Answer: D

Chart Type: *Area Chart*

Difficulty: *High*

Explanation: *The chart's use of a continuous area encoding across unordered categories creates a false impression of trend or progression. While the uniform color may affect readability, it is not the primary source of the misleading interpretation.*

Figure 12: An example question from the **Manipulated Visual Encoding** group, categorized under *Mismatched Encoding: Continuous encoding for categorical data* and presented as a *Area Chart*.



The chart displays the number of working days lost over several years due to industrial actions. Which of the following statements is accurate?

A: The number of working days lost in 2011-12 is around four times that lost in 2010-11.
B: The number of working days lost in 2007-08 is the highest among all the years shown.
C: The number of working days lost in 2006-07 is the lowest among all the years shown.
D: The number of working days lost in 2008-09 are more than 2009-10.

Correct Answer: C

Misleading Answer: A

Misleader: *Dual Encoding*

Chart Type: *Bar Chart*

Task: *Data Comparison*

Difficulty: *Low*

Explanation: *Although the actual value for 2011–12 is about 1.8 times greater than 2010–11, the bar appears nearly four times larger due to dual encoding—the bar's height, width, and color intensity all simultaneously represent the same data, exaggerating the perceived difference.*

Figure 13: An example question from the **Manipulated Visual Encoding** group, categorized under *Dual Encoding* and presented as a *Bar Chart*.

A.11 Prompt Templates

A.11.1 Automated MCQ Expansion and Iterative Refinement workflow

The following are the prompts for each components in the proposed Automated MCQ Expansion and Iterative Refinement workflow (fig. 3).

Chart Variation

Generate HTML Variation

You are generating misleading HTML-based charts for a QA benchmark using D3.js. The goal is to modify the visualization to reflect the misleader {*misleader*} by adjusting the chart's visual representation while maintaining core structure and labels.
Requirements:
1. The base HTML provided serves as the primary reference. Maintain the same overall structure, styles, and chart components. The generated HTML must be directly runnable.
2. Retain the following from the base HTML:
 - Chart dimensions (fixed at 1000x750 pixels).
 - Titles, legends, axis labels, and grid lines.
 - D3.js visualization logic.
3. Modify the D3 chart to apply the misleader.
4. Ensure the chart reads data from the updated CSV path: {*csv_path_in_html*}.
 - Ensure there are no extra or duplicated closing parentheses ')' in the 'd3.csv' function call.
5. Prevent overflow by adjusting margins and ensuring all chart elements fit within the canvas.
6. Use the labelled JPEG sample as a visual guide to ensure the misleader effect is accurately represented.
7. Remove all unnecessary comments, such as:
 - Descriptive comments like "Here's the complete and executable HTML page..."
 - Markdown syntax (e.g., “‘html, “‘).
8. **Ensure the chart title reflects the new chart topic but do not infer the misleader in the chart title**:
 - The title should match the description of the relevant CSV columns. Make sure do not infer the misleaders in chart title. Keep the same
9. **Ensure axis labels dynamically update**:
 - Use the column names from the CSV data for axis labels whenever appropriate. Make sure do not infer the misleaders in the axis labels.
Returns: str: The generated HTML content only.

Misleader: {*misleader*}
Misleader Description: {*misleader_description*}
Chart Type: {*chart_type*}
CSV Data (Driving the Chart): {*csv_data*}
Base HTML (Reference for Structure and Style): {*base_html*}
JPEG (Labelled Misleader):
 - Refer to the attached JPEG for visual alignment. Path to JPEG: {*jpeg_path*}
Ensure the full visualization code (chart headings, legends, titles, axes) is preserved:
Return the output as a complete and executable HTML page in the following format:

~~~
 ```
<!DOCTYPE HTML>
<html lang="en">
<head>
    <meta charset="UTF-8">
   <meta name="viewport" content="
    width=device-width,
    initial-scale=1.0">
    <script src="https://d3js.org/
    d3.v6.min.js"></script>
    <style>
        #chart {{
            width: 1000px;
            height: 750px;
            margin: 60px auto;
        }}
        .axis path, .axis line {{
            stroke: black;
        }}
        .dot {{
            fill: steelblue;
            stroke: black;
            stroke-width: 1px;
        }}
        .avg-line {{
            stroke: black;
            stroke-dasharray: 4,4;
        }}
        .annotation {{
            font-size: 12px;
~~~

```
                font-weight: bold;
                fill: black;
            }}
        </style>
    </head>
    <body>
      <h1> // // Insert appropriate chart
        heading like the base HTML,
      ensure don't disclose the misleader
        information here </h1>
        <div id="chart"></div>
        <script>
          // Insert D3.js visualization
      logic extracted from base HTML here
        </script>
    </body>
    </html>
      ```
```

- Ensure that the returned HTML page preserves the full chart functionality and visualization logic from the base HTML.

- Implement the misleader described above by modifying axis scaling, bar order, or annotation placement.

- The goal is to introduce subtle distortions that create misleading visual interpretations while retaining the core chart layout.

Generate CSV Variation

You are modifying CSV data for a $\{chart\_type\}$ visualization that reflects the misleader $\{misleader\}$.
Instructions:
1. Keep the same number of columns ($\{expected\_num\_columns\}$) as the original CSV.
2. Ensure each column has the same data type (e.g., int, float, string) as the original CSV.
3. Modify column names and data values to reflect the misleader effect:
- $\{misleader\_description\}$
4. Return only the modified CSV content with no additional comments or metadata.

Original CSV Data: $\{csv\}$

QA Generation

Generate QA Variation

You are generating Q&A content for a misleading chart which is generated as a variation of the sample example. Please strictly follow the style of the sample (in which a chart with labeled misleading region and the corresponding Q&A is provided). The goal is to craft a question that highlights the misleading aspect of the variation chart accordingly.

Requirements:
1. Follow the structure of the provided JSON file exactly.
2. Frame the question to reflect the misleading aspect of the chart.
3. Adjust the options (A, B, C, D) to ensure one option aligns with the misleader.
4. Indicate the correct answer clearly.
5. Choose the most misleading option as "wrongDueToMisleaderAnswer" to highlight the most plausible incorrect option caused by the misleading chart.
6. Reference the JPEG-labelled chart and Q&A sample to ensure the explanation correctly addresses the visual misleader.
7. Set the "ifLabelled" field to "False" to indicate the chart is not labelled.

Misleader: $\{misleader\}$
Misleader Description: $\{misleader\_description\}$
Chart Type: $\{chart\_type\}$
CSV Data (Driving the Variation Chart): $\{csv\_data\}$
The target Misleading Chart (Variation Chart): $\{chart\_variation\}$
Sample Q&A JSON (Structure Reference): $\{base\_json\}$
Sample Chart JPEG (with Labelled Misleader):
- Refer to the attached JPEG for visual alignment.
- Path to JPEG: $\{jpeg\_path\}$
Return the output in this strict format:

````
   ```json
  {{
````

~~~
    "question": "Based on the chart,
  what is the approximate average sales
    for Q1 2023 in Restaurant X?",
    "options": {{
      "A": "120",
      "B": "180",
      "C": "220",
      "D": "250"
    }},
    "correctAnswer": "B",
    "misleader": "{misleader}",
    "chartType": "{chart_type}",
    "task": "Aggregate Values",
  "explanation": "The chart annotation
  shows 'Reference: 220', but the true
    average is 180. Misleading
    annotations cause users
    to misjudge the data.",
    "difficulty": "Medium",
    "ifLabelled": "False",
    "wrongDueToMisleaderAnswer": "C"
  }}
  ```
~~~

Automated Evaluation & Feedback & Refinement Loop

Variation Evaluation

You are tasked with evaluating and refining a visualization QA sample for a misleading chart.

** Inputs **

- **QA Content**: $\{qa\_content\}$
- **Misleader Description**: $\{misleader\_desc\}$
- **Misleadering Chart Image**: $\{chart\_image\}$
- **CSV Variation Check**: $\{csv\_variation\_status\}$
- **Generated CSV **: $\{generated\_csv\}$
- **Original CSV **: $\{original\_csv\}$

** Task **
Evaluate the chart (visualization), question, QA options, correct answer, wrong-Due-To-Misleader-Answer all match the misleader description. If you find anything wrong, try to identify the corresponding errors in the CSV, QA, and HTML components based on the below guidelines and commen issues.
Ensure:

- Make sure to double check the visualization indeed represents the intended misleader as described in the misleader description!
- Make sure to check if the QA content matches the misleader and visualization.
- Make sure to double check the correctness of the correct answer and wrongDueToMisleaderAnswer based on the misleader description and the chart figure!
- Make sure to check if the generated CSV introduces meaningful variations compared to the original CSV.
- Make sure to double check the items in the list of "Some common issues include" below.

** Guidelines **
Evaluate the chart (visualization), question, QA options, correct answer, wrong-Due-To-Misleader-Answer, and alignment with the misleader description. Provide status as 'correct' or 'incorrect':

- "correct": No refinement needed.

- "incorrect": Refinement needed, provide comments and instructions.

- If the sample is correct, set "status": "correct" and leave "comments", "revision_instructions", and "updated_content" fields empty or as "No issues" and "null".
- If the sample requires refinement, set "status": "incorrect" and provide detailed comments and specific revision instructions for each component ("csv", "qa", "html").

** For the updated_content for "qa", directly provided the revised content in JSON format. **
** For the updated_content for "csv" and "HTML", provide very detailed samples and do not include the whole code. **

** Some common issues include: **
CSV:
- The data values have no changes (no small variations) with the original data. Only changed the column names.
- Incorrect or missing data values.

QA:
- Mismatched question context (e.g., question does not align with the chart's content).
- Mismatched options (e.g., no correct answer choices exist).
- Missing or incorrect correct answers (e.g., no correct option, or wrong answer marked as correct).
- Incorrect explanations (e.g., explanation does not match the chart or the misleader description).
- Incorrect or missing wrongDueToMisleaderAnswer (e.g., wrong answer does not align with the misleader).

HTML:
- The CSV data path in the D3.js code is incorrect. Ensure the path in the D3.js code is path: $\{csv\_path\_in\_html\}$.
- Disclose the misleader in the visualization title (e.g., title implies it is a misleading visualization).
- Not specified by misleader description, but still missing labels or legend.
- Have any annotations to indicate misleading nature. Need to remove them.
** Output Format **
Return a JSON object with the following structure:

~~~
```json
   {{
     "status": "<correct/incorrect>",
       "comments": {{
           "csv": "<Comment for CSV
         refinement or 'No issues'>",
           "qa": "<Comment for QA
         refinement or 'No issues'>",
           "html": "<Comment for HTML
          refinement or 'No issues'>"
       }},
       "revision_instructions": {{
           "csv":
           "<Specific instructions
           for revising the CSV or
           'No revision required'>",
           "qa":
           "<Specific instructions
           for the revised QA or
           'No revision required'>",
           "html":
           "<Specific instructions
           for revising the HTML or
           'No revision required'>"
       }},
       "updated_content": {{
           "csv_data": "<Updated CSV
       content if applicable or null>",
           "qa_content": "<Updated QA
       content if applicable or null>",
           "html_content": "<Updated
           HTML content if applicable
           or null>"
       }}
   }}
```
~~~

Revision Loop: CSV

You are tasked with revising a CSV file to address specific issues. If you find no issues mentioned in the Comments and Instructions or they are unclear, please directlty output the Current CSV Content {*csv_content*} without any changes.

*** Comments:
{*comments*}

*** Instructions:
{*instructions*}

*** Current CSV Content:
{*csv_content*}

*** Revised CSV Sample:
{*revised_csv_sample*}

*** Task
Make the necessary revisions to the CSV file according to the Comments, Instructions and Revised CSV Sample. Return the updated content as a valid CSV file.

Revision Loop: HTML

You are tasked with revising an HTML file to address specific issues. If you find no issues mentioned in the Comments and Instructions or they are unclear, please directlty output the Current HTML Content {*html_content*} without any changes.

*** Comments:
{*comments*}

*** Instructions:
{*instructions*}

*** Current HTML Content:
{*html_content*}

*** Task
Make the necessary revisions to the HTML file and return the updated content as valid and executable HTML.

Ensure the full visualization code (chart headings, legends, titles, axes) is preserved:

Make sure to replace the CSV path in the D3.js code with the correct path {*csv_path_in_html*}.

Make sure to remove any annotations or titles in the visualization that disclose the misleader! (e.g., should not have some extra titles indicating the potential misleader)

Make sure the visualization represents the misleader as intended.

Make sure to not change the other parts of the visualization code.

Return the output as a complete and executable HTML page in the following format:

~~~~
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content=
    "width=device-width,
    initial-scale=1.0">
    <script src="https://d3js.org/
    d3.v6.min.js"></script>
    <style>
~~~~

~~~
        #chart {{
            width: 1000px;
            height: 750px;
            margin: 60px auto;
        }}
        .axis path, .axis line {{
            stroke: black;
        }}
        .dot {{
            fill: steelblue;
            stroke: black;
            stroke-width: 1px;
        }}
        .avg-line {{
            stroke: black;
            stroke-dasharray: 4,4;
        }}
        .annotation {{
            font-size: 12px;
            font-weight: bold;
            fill: black;
        }}
    </style>
</head>
<body>
   <h1> // Insert appropriate chart
    heading like the base HTML,
    ensure do not
    disclose the misleader
    information here </h1>
    <div id="chart"></div>
    <script>
       // D3.js visualization logic
      d3.csv("{csv_path_in_html}")
           .then(function(data) {{
               // Chart logic here
           }})
         .catch(function(error) {{
              console.error('Error
               loading CSV data:',
                error);
           }});
    </script>
</body>
</html>
```
~~~

Revision Loop: Q&A

You are tasked with revising a QA JSON file to address specific issues. If you find no issues mentioned in the Comments and Instructions or they are unclear, please directlty output the Current QA Content *{qa_content}* without any changes.

*** Comments:
{comments}

*** Instructions:
{instructions}

*** Current QA Content:
{qa_content}

*** Revised QA Recommendation:
{revised_qa_recommendation}

*** Task
Make the necessary revisions to the QA JSON file and return the updated content as valid JSON.
Return the output in this strict format:

~~~
```json
{{
   "question": "Based on the chart, what
    is the approximate average sales for
     Q1 2023 in Restaurant X?",
     "options": {{
     "A": "120",
     "B": "180",
     "C": "220",
     "D": "250"
     }},
     "correctAnswer": "B",
     "misleader": "misleader",
     "chartType": "chart_type",
     "task": "Aggregate Values",
    "explanation": "The chart annotation
    shows 'Reference: 220', but the true
 average is 180. Misleading annotations
     cause users to misjudge the data.",
     "difficulty": "Medium",
     "ifLabelled": "False",
    "wrongDueToMisleaderAnswer": "C" }}
```
~~~

A.11.2 Prompt Templates for the Main Experiments

The following are the prompt templates for the **Baseline** and **Zero-shot CoT** experimental settings (table 1).

Baseline

Core Prompts for Baseline Experiment

You are given a potentially misleading chart and a multiple-choice question related to it. Please provide the MCQ answer and the corresponding explanation:

** The Potentially Misleading Chart: ** $\{image\_path\}$

** Question: ** $\{question\}$

** Options: ** $\{formatted\_options\}$

** Instructions: **

- **Only output the selected option on the first line (A, B, C, or D).**
- Then, on a new line, **provide a detailed explanation** on why this choice is correct based on the chart.

- Your response format must strictly follow:
  <Letter Choice>
  <Explanation>
- For example:

```
B
The price trend is decreasing from
 1975 to 1980, as the line clearly
  slopes downward.
```

Now, answer accordingly, do not forget to provide the explanation for your answer:

Zero-shot CoT

Core Prompts for Zero-shot CoT Experiment

You are given a potentially misleading chart and a multiple-choice question related to it. Please provide the MCQ answer and the corresponding explanation. ** Let's think and solve the question step by step!**

** The Potentially Misleading Chart: ** $\{image\_path\}$

** Question: ** $\{question\}$

** Options: ** $\{formatted\_options\}$

** Instructions: **

- **Start with breaking down the problem and think through the question logically.
- **You can first try to analyze the chart components (e.g., chart title, chart axis, ...), then based on the chart analysis, continue with the analysis of QA.
- After reasoning, output the selected option (A/B/C/D) and explain your choice based on the chart.

** Please Ensure: **

- **Only output the selected option on the first line (A, B, C, or D).**
- Then, on a new line, **provide a detailed explanation** on why this choice is correct based on the chart.
- Your response format must strictly follow:
  <Letter Choice>
  <Explanation>
- For example:

```
B
The price trend is decreasing from
 1975 to 1980, as the line clearly
  slopes downward.
```

Now, answer accordingly, do not forget to provide the explanation for your answer:

A.11.3 Region-Aware Misleading Chart Reasoning Pipeline

The following are the prompts for each components in the proposed Region-Aware Misleading Chart Reasoning pipeline (fig. 4).

Misleading Region Identification

MLLM Module for Misleading Region Identification

You are given a chart (dimensions: 2400 x 2122) with potential misleading regions: $\{image\_path\}$

Please analyze the image to detect any misleading regions (e.g., the chart design or data select might be intentionally manipulate the data's visual representation to bolster specific claims, can distort viewers' perceptions and lead to decisions rooted in false information).

** Let's think it step by step! ** Here is a potential checklist for identifying misleading regions that you may refer to:

- Chart Title
- Chart Type
- X and Y Axis
- Chart Legend
- Chart Visual Encoding
- Chart Data Use and Choice
- Chart Scales
- Chart Annotations

Then output a JSON file containing coordinates for the potential misleaders and explanations.

*** Instructions: - **Please analyze the image (dimensions: 2400 x 2100) to detect any misleading regions.**

- **Provide the misleading region coordinates with a detailed explanation**

- Your response format must strictly follow the example JSON format:

```
```
[
 {{"coordinates": [[100, 200],
   [150, 200],[100, 300],
   [150, 300]],
   "explanation": "The chart
   incorrectly scales
   the y-axis."}},
 {{"coordinates": [[250, 300],
   [300, 300],[250, 350],
   [300, 350]],
 "explanation": "The chart uses
   misleading colors that
   misrepresent data."}}
]
```
```

Q&A with Labeled Reference Region

MLLM Module for Q&A with Labeled Reference Region

You are given a chart with potential misleading regions and a corresponding question. Additionally, you will receive an extra image where the potential misleading region is labeled with an explanation. Use this as a reference, ** but please note that the labels may not always be accurate! ** Answer the question with a clear explanation.

** The original Chart: ** $\{image\_path\}$

** Question: ** $\{question\}$

** Options: ** $\{formatted\_options\}$

** The labeled Chart: ** $\{labeled_{i}mage\_path\}$

** Explanations for the labels: ** $\{regions\_explanation\}$

** Instructions: **

- **Only output the selected option on the first line (A, B, C, or D).**

- Then, on a new line, **provide a detailed explanation** on why this choice is correct based on the chart.

- Your response format must strictly follow:
  <Letter Choice>
  <Explanation>
- For example:

```
```
B
The price trend is decreasing from
```
```

~~~~
      1975 to 1980, as the line clearly
        slopes downward.
        ```
~~~~

Now, answer accordingly: