PolyChartQA: Benchmarking Large Vision-Language Models with Multilingual Chart Question Answering

Anonymous ACL submission

Abstract

Charts are a universally adopted medium for data communication, yet existing chart understanding benchmarks are overwhelmingly English-centric, limiting their accessibility and relevance to global audiences. To address this limitation, we introduce **PolyChartQA**, the first large-scale multilingual benchmark for chart question answering, comprising 22,606 charts and 26,151 QA pairs across 10 diverse languages. PolyChartQA is built via a novel pipeline that enables scalable multilingual chart generation through data translation and code reuse, incorporating state-of-the-art LLM-based translation and rigorous quality control. We systematically evaluate multilingual chart understanding with PolyChartQA on state-of-the-art LVLMs and reveal a significant performance gap between English and other languages, particularly low-resource ones, which exposes a critical shortcoming in current models. To mitigate this gap, we introduce a large-scale training dataset, **PolyChart-Instruct**, containing 751,363 multilingual chart QA pairs across 131,515 chart images. Fine-tuning Qwen2.5-VL with PolyChart-Instruct yields average performance gains of up to 14 points. Together, our benchmark and datasets provide a foundation for developing globally inclusive vision-language models capable of understanding charts across diverse linguistic contexts.

1 Introduction

Charts are ubiquitous tools for visualizing quantitative information and supporting analytical reasoning across domains such as science, business, and journalism. Accurate chart interpretation is essential for data-driven decision-making. Recent advances in large vision-language models (LVLMs) have enabled significant progress in perceiving and reasoning over visualizations such as plots, diagrams, and charts. These models have shown promising results on tasks including complex chart question answering (Masry et al., 2022; Xia et al., 2024; Wang et al., 2024c; Masry et al., 2025), chart summarization (Rahman et al., 2022; Tang et al., 2023), and chart image re-generation (Moured et al., 2024; Yang et al., 2024).

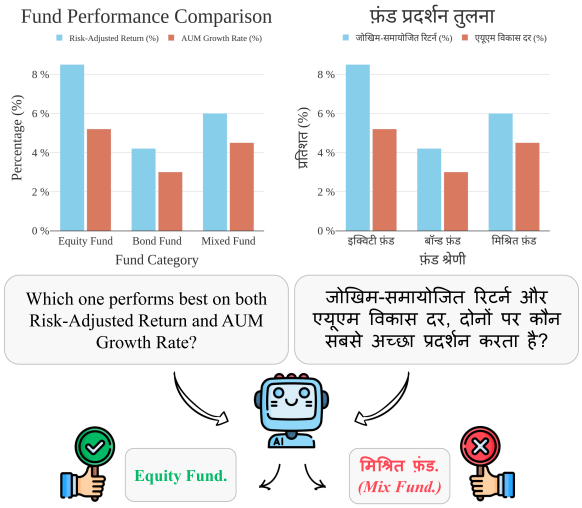


Figure 1: Example of inconsistent chart understanding by LVLMs. The model answers correctly in English but fails on the Hindi equivalent.

However, existing benchmarks for chart understanding remain overwhelmingly English-centric, overlooking the unique challenges of multilingual comprehension. As shown in Figure 1, leading LVLMs often succeed on English chart QA but struggle with their translated counterparts. This English-only bias poses a major barrier to developing globally inclusive chart understanding models, especially for underrepresented languages. While recent works (Chen et al., 2024a; Heakl et al., 2025) have introduced bilingual chart datasets, they remain limited in scale and language coverage. To date, **no** comprehensive benchmark exists for evaluating multilingual chart understanding in LVLMs.

Moreover, existing multilingual multimodal benchmarks (Pfeiffer et al., 2021; Liu et al., 2021; Yu et al., 2025; Liu et al., 2024b; Xuan et al., 2025)

primarily focus on natural images, with limited attention to structured data like charts. Although M3Exam (Zhang et al., 2023) and xMMMU (Yue et al., 2024) include a few chart-based questions in scientific subjects, they are not dedicated to chart understanding and lack coverage of holistic reasoning. A key obstacle to building such benchmarks is the high cost of multilingual chart image annotation (Romero et al., 2024; Tang et al., 2024), which limits scalability.

To overcome these challenges, we introduce a **scalable two-stage pipeline**. In the first stage, we generate high-quality English data by decoupling any chart into a structured JSON and a code template. In the second stage, we apply state-of-the-art LLMs to translate chart data and QA pairs, followed by automatic multilingual rendering and **multi-stage quality control**, including both automated checks and human review.

Using this pipeline, we construct **PolyChartQA**, the first large-scale benchmark for multilingual chart understanding, comprising over 22K chart images and 26K QA pairs in ten widely spoken languages: English, Chinese, Hindi, Spanish, French, Arabic, Bengali, Russian, Urdu, and Japanese, collectively spoken by over 65% of the global population. Our benchmark includes both real-world and synthetic charts, offering a diverse and rigorously curated resource for evaluating and advancing multilingual chart understanding.

Using PolyChartQA, we conduct the first systematic evaluation of multilingual chart understanding in LVLMs. Experimental results reveal two major findings: (1) Current models demonstrate limited effectiveness on multilingual chart QA tasks, with particularly low performance in low-resource languages. (2) Significant performance gaps persist across different scripts and language families, suggesting that existing models lack cross-lingual generalization.

To bridge this gap and enhance multilingual chart capabilities, we construct **PolyChart-Instruct**, a large-scale training set with 751K instruction QA pairs across 131K charts. Fine-tuning Qwen2.5-VL with a two-stage instruct-tuning approach on PolyChart-Instruct yields substantial gains, with average accuracy improvements of up to 13 points, demonstrating the effectiveness of instruction tuning for multilingual chart reasoning. In summary, our main contributions are:

- We propose a **unified and reproducible pipeline** for building high-quality, large-scale multilingual chart QA datasets, leveraging LLM-based translation and code-driven chart generation.
- We present **PolyChartQA**, the first benchmark to enable systematic evaluation of LVLMs on chart understanding in ten diverse languages, along with a large-scale new training dataset **PolyChart-Instruct**.
- We conduct **extensive empirical studies** that offer critical insights into current performance gaps, while demonstrating how our proposed datasets significantly bridge these gaps, especially for low-resource languages.

2 Related Work

2.1 Chart Understanding Datasets

Chart understanding tasks challenge models to interpret both visual and textual information within charts and to provide accurate responses to a range of instructions. In recent years, several benchmarks have been introduced to systematically evaluate the capabilities of Large Vision-Language Models (LVLMs) across tasks such as chart question answering (Masry et al., 2022; Methani et al., 2020; Kantharaj et al., 2022a), chart summarization (Tang et al., 2023; Kantharaj et al., 2022b; Rahman et al., 2022), chart-to-table conversion (Xia et al., 2023, 2024; Chen et al., 2024a), and chart re-rendering (Moured et al., 2024; Yang et al., 2024). Of these, chart question answering has emerged as a central metric for assessing a model's ability to perform fine-grained chart understanding.

Early datasets such as FigureQA (Kahou et al., 2017), DVQA (Kafle et al., 2018) and PlotQA (Methani et al., 2020) featured synthetic charts and template questions, limiting their diversity and real-world applicability. More recent benchmarks, including ChartQA (Masry et al., 2022), ChartX (Xia et al., 2024), MMC (Liu et al., 2023), and ChartXiv (Wang et al., 2024c), incorporate real-world charts and human-authored questions, broadening the range of chart types and question complexities represented in the evaluation.

Despite recent advances, most chart datasets are English-only, with limited multilingual benchmarks (Chen et al., 2024a; Heakl et al., 2025). This lack of coverage prevents comprehensive evaluation of LVLMs on multilingual chart understanding and limits their real-world applicability.

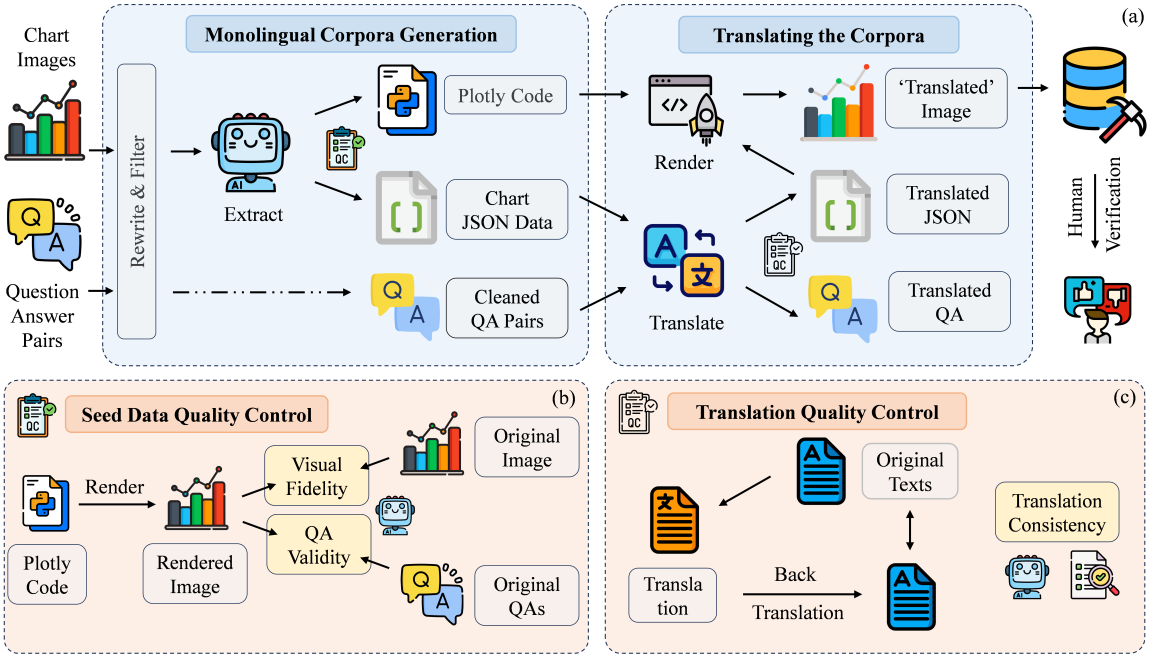


Figure 2: Overview of the PolyChartQA data pipeline. (a) The full workflow consists of two stages: monolingual corpus construction and multilingual translation. (b) Quality control procedures applied with seed data generation. (c) Quality control procedures applied during the translation stage.

2.2 Multilingual LVLMs

Building on the progress of foundational monolingual models (Li et al., 2023; Team et al., 2024a,b), researchers have developed a wide range of LVLMs with multilingual capabilities. Early influential works include PaLI (Chen et al., 2022), mBLIP (Geigle et al., 2023), and PaliGemma (Beyer et al., 2024; Steiner et al., 2024), which pioneered scalable multilingual vision-language alignment.

More recently, open-source models such as PALO (Maaz et al., 2024), Maya (Alam et al., 2024), Pangea (Yue et al., 2024), and Centurio (Geigle et al., 2025) have significantly broadened linguistic coverage and improved cross-lingual visual understanding. In parallel, open-source families like QwenVL (Bai et al., 2023, 2025; Wang et al., 2024b), InternVL (Chen et al., 2024b,c,d), and Phi-Vision (Abdin et al., 2024a,b) have demonstrated strong performance on multilingual multimodal tasks. Despite these advances, the ability of these models to process complex, text-rich visual inputs such as charts in multiple languages remains an underexplored challenge.

2.3 Multilingual Evaluations on LVLMs

The rapid development of multilingual LVLMs has driven the creation of a diverse set of benchmarks to assess their performance across multimodal tasks. These benchmarks typically cover general cross-lingual VQA (Pfeiffer et al., 2021; Changpinyo et al., 2022), text-centric VQA (Tang et al., 2024; Yu et al., 2025), and culturally diverse VQA (Romero et al., 2024; Liu et al., 2021; Vayani et al., 2024). In addition to task-specific resources, several comprehensive evaluation suites have been proposed to measure broader multilingual and multimodal capabilities of LVLMs. Benchmarks such as MMBench (Liu et al., 2024b), MMLU-Prox (Xuan et al., 2025), and M4U (Wang et al., 2024a) span a wide array of tasks including multimodal reasoning, open-domain chat, image captioning, and math problem solving. Similarly, M3Exam (Zhang et al., 2023) and Exams-V (Das et al., 2024) offer large-scale, real-world exam-style evaluations for LVLMs in multilingual and multimodal settings. Despite recent progress, most benchmarks overlook structured data like charts. While datasets such as M3Exam (Zhang et al., 2023) and SMPQA (Geigle et al., 2025) include chart-related content, they remain limited in scale and task diversity, focusing more on OCR than holistic reasoning.

3 Building PolyChartQA and PolyChart-Instruct

We present **PolyChartQA**, a large-scale multilingual chart question answering benchmark de-

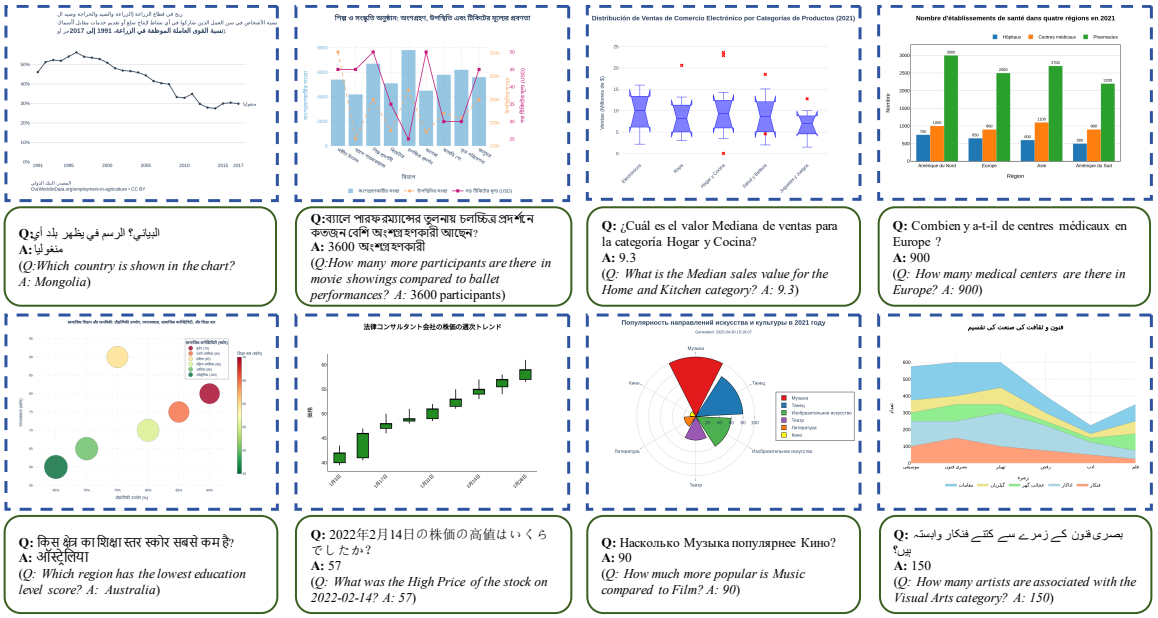


Figure 3: Multilingual chart question answering visualizations selected from POLYCHARTQA. First row, from left to right: Arabic, Bengali, Spanish, French. Second row, from left to right: Hindi, Japanese, Russian, Urdu.

signed to address the lack of multilingual resources in this field; and **PolyChart-Instruct**, a large-scale training corpus for multilingual chart understanding. Both of the datasets cover ten languages: English, Chinese, Hindi, Spanish, French, Arabic, Bengali, Russian, Urdu, and Japanese. Figure 2 illustrates the unified data pipeline we adopt to construct both datasets. It begins with the selection and refinement of high-quality English chart corpora and systematically expands to other languages via LLM-based translation and rigorous quality control. In what follows, we first detail the pipeline design in Section 3.1, then introduce the construction of the evaluation benchmark in Section 3.2, and finally describe the training dataset in Section 3.3.

3.1 Data Construction Pipeline

3.1.1 Monolingual Corpora Construction

English Dataset Selection. We began by surveying publicly available English-language chart QA benchmarks, evaluating each based on three key criteria: (i) overall data quality and chart diversity; (ii) the breadth and clarity of question types; and (iii) demonstrated adoption within the research community.

To balance realism, diversity, and usability, we selected ChartQA (Masry et al., 2022), ChartLlama (Han et al., 2023), and ChartX (Xia et al., 2024) as the source dataset to construct our multilingual chart datasets.

Data Cleaning and Validation. We applied a two-step quality control pipeline to the datasets.

First, each image–question–answer triplet was automatically checked with *Gemini 2.5 Pro* to verify whether the answer could be reliably inferred from the chart given the question. Flagged data items are either corrected by human experts or discarded. In the second stage, we manually normalized answers that were excessively long or verbose, standardizing them into concise formats—such as single words, numbers, or short phrases—while preserving their original semantics to make fair evaluation.

To assess residual noise, we randomly sampled 10% of the cleaned dataset for human review and achieved a pass rate exceeding 98%, confirming the corpus's low noise and high consistency for downstream multilingual expansion.

Seed Data Generation. A central innovation of our pipeline is the decoupling of chart content and visual rendering. For each cleaned chart, we prompted *Gemini 2.5 Pro* to generate two key artifacts: a structured JSON file that encodes the data table, chart type, text layout, color scheme, and other essential visual attributes, and an executable Python script that reconstructs the chart using *Plotly*. Special attention was given to multilingual text rendering, especially for right-to-left (RTL) languages and other non-Latin scripts. This decoupled design facilitates the direct ingestion and regeneration of arbitrary chart images, improving

both compatibility and reusability.

3.1.2 Translating the Corpora

Text Data Translation. While standard machine translation services often struggle to preserve the structure and semantics of chart JSON files and their associated QA pairs, recent studies (Qiu et al., 2022; Chen et al., 2023; Maaz et al., 2024) have shown that LLM-based translation offers significantly better fidelity and consistency. Building on this insight, we adopt an LLM-based workflow using *Gemini 2.5 Pro*, which jointly translates the chart JSON data along with their corresponding questions and answers, ensuring coherence between the translated chart content and the QA texts.

We explicitly instruct the model to maintain semantic equivalence and adhere to the target language's cultural conventions, thereby mitigating translation errors. Empirical observations suggest that the translated corpora exhibit reasonably high levels of semantic and structural alignment with the original English data, supporting reliable cross-lingual evaluation and scalable multilingual benchmarking.

Multilingual Data Generation. The translated JSON files are then paired with the corresponding template code to generate chart images in each target language. As in the monolingual stage, we discard any samples for which the code fails to execute successfully.

3.1.3 Quality Control

Our pipeline incorporates a multi-stage quality control mechanism to ensure both the accuracy and usability of the constructed dataset. These quality control procedures are applied at different stages for the English seed data and its multilingual extensions. We will describe the quality control strategies in details below.

Seed Data Quality Control. To ensure the integrity of the seed dataset, we implement a two-stage validation process shown in Figure 2 (b). **(i) Visual Fidelity**: We assess the visual fidelity of each regenerated chart by comparing it to the original, using *Gemini 2.5 Pro* to detect semantic and stylistic discrepancies. Charts exhibiting significant inconsistencies in type, values, or layout are removed. **(ii) QA Validity**: We verify that all questions remain answerable on the reconstructed charts, using *Gemini 2.5 Pro* and *GPT-4.1* as independent validators. Only samples confirmed as valid by both models are retained, effectively reducing instances with potential semantic drift or poor linguistic quality.

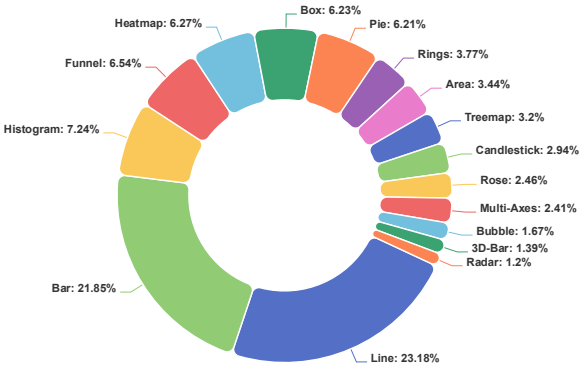


Figure 4: Distribution of chart types in PolyChartQA.

Multilingual Data Quality Control. Building upon the validated seed data, we apply a dedicated quality control procedure to the multilingual outputs. This protocol is composed of two primary stages designed to ensure both textual and visual fidelity. **(i) Translation Quality**: We first conduct automated validation for all translated text, as illustrated in Figure 2 (c). Each instance is back-translated into English and compared against its original version. Consistency is evaluated using the METEOR (Banerjee and Lavie, 2005) metric and supplemented with semantic judgments from *Gemini 2.5 Pro*. Samples with low alignment or poor linguistic quality are removed. **(ii) Visual Inspection**: All remaining multilingual chart images are manually reviewed to identify and discard those with visual defects such as text clipping, layout misalignment, or rendering errors. This step ensures high visual quality in the final dataset.

3.2 PolyChartQA

3.2.1 Data Statistics

We construct **PolyChartQA** by applying the above pipeline to the test sets of ChartQA (Masry et al., 2022) and ChartX (Xia et al., 2024). To the best of our knowledge, this is the first multilingual benchmark specifically designed for chart question answering. The dataset contains 22,606 chart images and 26,151 corresponding QA pairs, spanning 10 languages. PolyChartQA covers 16 chart types with a balanced distribution, as illustrated in Figure 4. Representative examples are shown in Figure 3. More detailed statistics of PolyChartQA are provided in Appendix A.1.

3.2.2 Human Evaluation

To further assess the quality of our PolyChartQA benchmark, we conducted a comprehensive human evaluation on a randomly sampled 20% subset for each language. Bilingual annotators were recruited to evaluate the data along three dimensions: **(i) Translation Quality**, which measures the semantic accuracy, fluency, and idiomatic appropriateness of the translated chart and QA pair, while ensuring that the translation introduces no bias, misinformation, or framing artifacts; **(ii) Chart Image Quality**, which assesses the visual clarity, textual legibility, and overall presentation of the translated chart; and **(iii) QA Correctness**, which evaluates whether the question is relevant to the chart and whether the provided answer is factually supported by the chart alone. Each aspect was rated on a three-point scale. Each instance was annotated by one annotator and independently reviewed by another to ensure reliability. Results in Table 1, demonstrate consistently high quality across all languages and dimensions.

Table 1: Human Evaluation Scores Across Nine Languages. Average Score is reported.

Language	Evaluation Aspects		
	Image	QA	Translation
Arabic	2.94	2.97	2.79
Urdu	2.71	2.92	2.71
Hindi	2.93	3.00	2.95
Bengali	2.91	2.98	2.92
Chinese	2.96	2.98	2.95
French	2.92	2.95	2.91
Spanish	2.84	2.95	2.87
Russian	2.65	2.71	2.92
Japanese	2.86	2.90	2.95

3.3 PolyChart-Instruct

3.3.1 Data Statistics

In addition to PolyChartQA, we construct **PolyChart-Instruct**, a large-scale fine-tuning dataset in the same way. multilingual instruction dataset intended for training purposes. It comprises 131,515 chart images and 751,363 corresponding QA pairs across 10 languages. This dataset follows the same multilingual generation pipeline as PolyChartQA with rigrious quality control. More detailed statistics of PolyChart-Instruct are provided in Appendix A.2.

4 Experiments

4.1 Evaluated LVLMs

To thoroughly assess the multilingual perception and reasoning abilities of modern LVLMs on our multilingual chart benchmark, we select representative state-of-the-art models from three categories: open-source general MLLMs, open-source multilingual LVLMs, and closed-source LVLMs.

The general open-source LVLMs include Qwen2-VL (Bai et al., 2023), InternVL 2.5 (Chen et al., 2024b), InternVL 3(Zhu et al., 2025), Phi-3 Vision (Abdin et al., 2024a), Phi-4 Multimodal (Abdin et al., 2024b), PaliGemma 2 (Team et al., 2024b), LLaVA-1.6 (Liu et al., 2024a), LLaVA-OneVision (Li et al., 2024), Llama-3.2-Vision (Grattafiori et al., 2024), and DeepSeek-VL2 (Wu et al., 2024). For open-source multilingual LVLMs, we evaluate PALO (Maaz et al., 2024), Maya (Alam et al., 2024), Pangea(Yue et al., 2024), and Centurio (Geigle et al., 2025). The closed-source category comprises Gemini-2.5-Pro (Comanici et al., 2025) and GPT-4o (Hurst et al., 2024). Closed-source models are accessed via their official APIs, while open-source models are run using their instruct versions available on the Hugging Face Model Hub.

4.2 Training Details

We adopt a two-stage supervised fine-tuning strategy based on Qwen2.5-VL. In the first stage, we perform multilingual alignment via a chart-to-JSON prediction task utilizing the chart metadata from PolyChart-Instruct.We further incorporate document and chart OCR tasks to enhance alignment, leveraging external datasets including MTVQA, PangeaOCR, and SMPQA (see Appendix C.3 for details). In the second stage, we fine-tune on our PolyChart-Instruct dataset for multilingual chart question answering.

We apply LoRA (Hu et al., 2022) in both stages with a fixed rank of 128. The alignment stage uses a learning rate of 5e-5 and unfreezes the vision encoder, while the instruction tuning stage uses a learning rate of 1e-5 with the vision encoder kept frozen. Each stage is trained for one epoch.

4.3 Evaluation Details

4.3.1 Implementation Details

All baseline models are evaluated under their official configurations. During inference, we set the decoding temperature to 0.01 and top_p to

Table 2: Overall performace on POLYCHARTQA benchmark. ‡ denotes the models fine-tuned on PolyChart-Instruct based on Qwen2.5-VL. Bold values in each model category denote the best performance.

Model	EN	ZH	FR	ES	RU	JA	AR	UR	HI	BN	Avg. (w EN)	Avg. (w/o EN)
Proprietary Models												
GPT-4o	55.9	46.0	53.4	54.4	52.4	45.4	50.5	48.7	51.3	48.2	50.9	50.2
Gemini-2.5-Pro	**70.6**	**67.7**	**69.0**	**69.3**	**67.6**	**68.6**	**69.1**	**67.5**	**68.6**	**66.0**	**68.5**	**68.2**
Open Source Models												
PaliGemma2-3B	26.6	14.7	19.7	21.5	13.9	10.7	15.9	12.2	14.3	10.2	16.3	14.9
Phi-3 Vision	45.1	17.5	37.2	36.9	26.9	15.7	9.3	4.7	10.6	10.6	23.2	20.2
InternVL-2.5-2B	27.8	3.3	14.7	9.2	9.5	2.0	4.3	0.3	1.2	0.1	7.8	5.1
InternVL-3-2B	43.7	35.3	30.8	33.5	25.6	26.9	17.1	14.6	15.7	11.9	25.6	23.1
Qwen2-VL-2B	42.3	33.6	37.6	37.7	35.9	22.2	28.8	19.1	24.4	23.0	30.7	29.1
Qwen2.5-VL-3B	**67.4**	**59.6**	**61.8**	**62.5**	**58.0**	**48.8**	**51.4**	**37.2**	**45.7**	**43.0**	**53.7**	**51.8**
LLaVA-OneVision-7B	18.7	10.1	13.1	14.2	9.4	8.3	7.5	5.2	7.1	5.7	10.1	9.0
LLaVA-v1.6-7B	24.8	12.9	18.9	18.2	13.5	11.5	12.0	7.7	10.0	6.7	13.9	12.4
Llama-3.2-Vision-11B	15.5	16.9	14.1	12.9	15.4	9.6	13.1	14.4	21.3	17.5	15.2	15.2
DeepSeek-VL2	40.1	38.8	26.4	34.1	19.9	0.0	14.2	13.8	19.1	16.3	24.8	22.5
InternVL-2.5-8B	39.2	26.3	32.4	33.5	29.5	22.6	10.9	11.2	14.0	13.4	23.5	21.4
InternVL-3-8B	54.1	39.4	43.4	45.8	38.1	39.7	21.4	17.2	20.2	17.5	33.8	31.0
Phi-4 Vision	**62.3**	46.0	55.9	44.6	48.7	41.6	29.7	23.4	33.4	18.3	40.6	37.7
Qwen2-VL-7B	56.4	54.3	53.4	52.7	52.2	47.3	40.5	32.0	43.9	40.3	47.3	46.1
Qwen2.5-VL-7B	60.5	**58.3**	**57.2**	**59.0**	**56.8**	**55.6**	**52.0**	**43.7**	**49.4**	**46.4**	**53.8**	**53.0**
Multilingual Models												
Centurio	7.9	4.0	3.6	3.0	1.5	2.5	2.0	1.5	1.5	1.0	2.9	2.2
Maya	8.7	6.4	7.6	7.2	6.8	6.0	7.1	5.7	6.9	5.6	6.8	6.6
PALO	11.5	6.0	10.5	9.9	7.0	5.9	7.0	5.0	5.2	3.6	7.3	6.7
Pangea	**24.7**	**13.6**	**19.8**	**21.3**	**15.8**	**11.5**	**13.1**	**12.1**	**13.1**	**13.1**	**16.1**	**14.9**
Fine-tuned Models‡												
Ours-3B	69.0	64.8	65.5	66.2	65.2	64.5	63.9	56.5	61.3	58.4	63.6	62.8
Δ over base model	+1.6	+5.2	+3.7	+3.7	+7.2	+15.7	+12.5	+19.3	+15.6	+15.4	+9.9	+11.0
Ours-7B	**73.7**	**69.2**	**71.3**	**70.7**	**68.0**	**68.0**	**66.1**	**62.7**	**66.6**	**62.9**	**68.0**	**67.2**
Δ over base model	+13.2	+10.9	+14.1	+11.7	+11.2	+12.4	+14.1	+19.0	+17.2	+16.5	+14.2	+14.2

0.7. We use a unified multilingual prompt: "Answer the question using a word or phrase in <target_language> or a number in digits. <Question>" All results are reported from a single run. Experiments are conducted on 8 NVIDIA A100 40GB GPUs.

4.3.2 Metrics

Following prior work (Masry et al., 2022), we adopt a type-aware relaxed accuracy metric: numerical predictions are considered correct if within 5% relative error of the ground truth; non-numerical answers require exact string match.

4.4 Evaluation Results

4.4.1 Zero-shot Evaluation

Table 2 presents zero-shot relaxed accuracy for a range of multilingual LVLMs on POLYCHARTQA. There is a clear and substantial performance gap between closed-source and open-source models. *Gemini-2.5-Pro* achieves the best results across all languages, with average accuracy reaching 0.685. In contrast, *GPT-4o* lags significantly behind, with an average accuracy of only 0.509.

Qwen2.5-VL-Series is the top performer among open-source models, consistently outperforming its peers in both high- and low-resource languages, even surpassing *GPT-4o*. By comparison, other open-source models (InternVL-Series, Phi-Series, etc.) show clear deficits, often struggling on non-English data.

General-purpose multilingual models such as Pangea-7B show limited effectiveness on chart QA, and the rest of this category perform even worse. This demonstrates that broad multilingual training alone does not equip models for structured visual

Table 3: Performance of Qwen2.5-VL 3B and 7B models under different cross-lingual conditions. Multi. Img. and Multi. QA denote whether the chart image or QA pair is multilingual. Bold values denote the best performance.

Model Size	Multi. Img.	Multi. QA	Avg. (w EN)	Avg. (w/o EN)
Qwen2.5-VL-3B	✗	✓	0.496	0.473
	✓	✗	0.521	0.499
	✓	✓	**0.537**	**0.518**
Qwen2.5-VL-3B	✗	✓	0.483	0.466
	✓	✗	0.510	0.495
	✓	✓	**0.538**	**0.530**

reasoning.

While all models excel in English, their effectiveness drops off sharply for lower-resource languages. High-resource languages such as Chinese and French remain relatively stable, but languages like Urdu and Hindi suffer pronounced declines. This highlights that current multilingual training pipelines fail to provide sufficient grounding in low-resource settings, likely due to data scarcity and unbalanced representation in the training corpora.

4.4.2 Training Results

As shown in Table 2, training on PolyChart-Instruct significantly boosts performance across both 3B and 7B models. The 3B variant achieves an average improvement of 11.0 points, while the 7B variant improves by 14.2 points, reaching performance comparable to *Gemini-2.5-Pro*.

In terms of per-language gains, All languages achieveconsistent performance gains from our training. Urdu benefits the most with up to 19.3 points for the 3B model and 19.0 points for the 7B model, followed by Bengali (+15.6 / +17.2) and Hindi (+15.4 / +16.5). These gains are particularly notable in low-resource languages, demonstrating the effectiveness of our training strategy. The results suggest that open-source LVLMs can substantially enhance their multilingual capabilities by incorporating high-quality, multilingual chart QA data.

4.4.3 Cross-lingual Evaluation

We evaluate cross-lingual performance on Qwen2.5-VL models by varying whether the chart image or QA pair is multilingual. Results in Table 3 show that using multilingual content in both modalities consistently yields the best accuracy, highlighting the benefit of aligning multilingual signals across vision and language. These findings also indicate that current LVLMs struggle to generalize across linguistic mismatches in visual content.

4.5 Ablation studies

4.5.1 Ablation on Training Strategies

As shown in Table 4, applying only fine-tuning already brings a significant 13.2 performance improvement compared to the baseline. Incorporating alignment training further enhances performance by 1 point.

Unfreezing the vision encoder during alignment while freezing it during fine-tuning yields the best performance (0.680), suggesting that early visual adaptation followed by stabilization enhances multilingual visual understanding. Ablation results of Qwen-2.5-VL-3B is shown in AppendixD.5.

Table 4: Performance of different training strategies on Qwen2.5-VL-7B. ❄ and 🔥 indicate that the vision encoder is frozen or unfrozen, respectively, during each stage. ✗ denotes that the stage is skipped. Bold values denote the best performance.

Training Strategy	Stage1	Stage2	Avg. (w EN)	Avg. (w/o EN)
Baseline	✗	✗	0.538	0.530
SFT only	✗	❄	0.669	0.661
	✗	🔥	0.670	0.662
Align+SFT	❄	❄	0.675	0.667
	🔥	❄	**0.680**	**0.672**
	🔥	🔥	0.671	0.663

5 Conclusion

In this paper, we present **PolyChartQA**, the first large-scale multilingual benchmark for chart question answering, spanning 22,606 charts and 26,151 QA pairs across 10 diverse languages. Built via a scalable pipeline, our dataset enables efficient and reproducible multilingual chart generation. Evaluation results reveal that current LVLMs struggle with multilingual chart understanding, particularly for non-Latin-script languages. To address this limitation, we introduce **PolyChart-Instruct**, a 750K instruction-tuning dataset that improves multilingual chart understanding by up to 14 points. We hope this work draws greater attention to the multilingual capabilities of LVLMs and serves as a foundation for developing more language-inclusive and globally accessible models.

Limitations

Despite introducing the first large-scale multilingual benchmark for chart question answering, PolyChartQA and PolyChart-Instruct still have limitations in language coverage. While it includes a diverse set of major languages, it excludes many lesser-spoken or low-resource ones, limiting its global inclusivity. Additionally, since PolyChartQA and PolyChart-Instruct build on existing datasets, it may inherit framing biases or inaccuracies from the source datasets. Although we employ a multi-stage validation process with partial human review, the use of LLM-based generation and translation may still introduce subtle shifts in tone, cultural framing, or emphasis across languages. Future work may explore fully human-annotated datasets when feasible and expand to more complex real-world visual formats such as infographics or interactive dashboards.

Ethics Statements

Our work aims to promote language inclusivity and accessibility in AI technologies by constructing a multilingual benchmark focused on chart understanding. By systematically evaluating model performance across diverse languages and scripts, especially those underrepresented in existing resources, we highlight current limitations and foster the development of more equitable large vision-language models. We believe this contributes to reducing the dominance of English in AI systems and supports the global community in accessing AI tools in their native languages. We acknowledge that our dataset, being derived from existing sources, may inherit biases or misinformation from the original charts. Furthermore, our use of LLMs for translation, despite a multi-stage validation process, may introduce subtle artifacts such as tonal shifts or cultural inaccuracies. We encourage future work to further improve multilingual data fidelity and broaden the linguistic inclusivity of AI systems.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, and 1 others. 2024a. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024b. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Nahid Alam, Karthik Reddy Kanjula, Surya Guthikonda, Timothy Chung, Bala Krishna S Vegesna, Abhipsha Das, Anthony Susevski, Ryan Sze-Yin Chan, S M Iftekhar Uddin, Shayekh Bin Islam, Roshan Santhosh, Snegha A, Drishti Sharma, Chen Liu, Isha Chaturvedi, Genta Indra Winata, Ashvanth. S, Snehanshu Mukherjee, and Alham Fikri Aji. 2024. Maya: An instruction finetuned multilingual multimodal model. *Preprint*, arXiv:2412.07112.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*.

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*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, and 1 others. 2024. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*.

Soravit Changpinyo, Linting Xue, Michal Yarom, Ashish V Thapliyal, Idan Szpektor, Julien Amelot, Xi Chen, and Radu Soricut. 2022. Maxm: Towards multilingual visual question answering. *arXiv preprint arXiv:2209.05401*.

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.

Nuo Chen, Zinan Zheng, Ning Wu, Ming Gong, Dongmei Zhang, and Jia Li. 2023. Breaking language barriers in multilingual mathematical reasoning: Insights and observations. *arXiv preprint arXiv:2310.20246*.

Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, and 1 others. 2022. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*.

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*.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, and 1 others. 2024c. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, and 1 others. 2024d. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Rocktim Jyoti Das, Simeon Emilov Hristov, Haonan Li, Dimitar Iliyanov Dimitrov, Ivan Koychev, and Preslav Nakov. 2024. Exams-v: A multi-discipline multilingual multimodal exam benchmark for evaluating vision language models. *arXiv preprint arXiv:2403.10378*.

Gregor Geigle, Abhay Jain, Radu Timofte, and Goran Glavaš. 2023. mblip: Efficient bootstrapping of multilingual vision-llms. *arXiv preprint arXiv:2307.06930*.

Gregor Geigle, Florian Schneider, Carolin Holtermann, Chris Biemann, Radu Timofte, Anne Lauscher, and Goran Glavaš. 2025. Centurio: On drivers of multilingual ability of large vision-language model. *arXiv preprint arXiv:2501.05122*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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*.

Ahmed Heakl, Abdullah Sohail, Mukul Ranjan, Rania Hossam, Ghazi Ahmed, Mohamed El-Geish, Omar Maher, Zhiqiang Shen, Fahad Khan, and Salman Khan. 2025. Kitab-bench: A comprehensive multi-domain benchmark for arabic ocr and document understanding. *arXiv preprint arXiv:2502.14949*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

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

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

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

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

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and 1 others. 2024. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

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

Fangyu Liu, Emanuele Bugliarello, Edoardo Maria Ponti, Siva Reddy, Nigel Collier, and Desmond Elliott. 2021. Visually grounded reasoning across languages and cultures. *arXiv preprint arXiv:2109.13238*.

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

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, and 1 others. 2024b. Mmbench: Is your multi-modal model an all-around

player? In *European conference on computer vision*, pages 216–233. Springer.

Muhammad Maaz, Hanoona Rasheed, Abdelrahman Shaker, Salman Khan, Hisham Cholakal, Rao M Anwer, Tim Baldwin, Michael Felsberg, and Fahad S Khan. 2024. Palo: A polyglot large multimodal model for 5b people. *arXiv preprint arXiv:2402.14818*.

Ahmed Masry, Mohammed Saidul Islam, Mahir Ahmed, Aayush Bajaj, Firoz Kabir, Aaryaman Kartha, Md Tahmid Rahman Laskar, Mizanur Rahman, Shadikur Rahman, Mehrad Shahmohammadi, Megh Thakkar, Md Rizwan Parvez, Enamul Hoque, and Shafiq Joty. 2025. Chartqapro: A more diverse and challenging benchmark for chart question answering. *Preprint*, arXiv:2504.05506.

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

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

Omar Moured, Sara Alzalabny, Anas Osman, Thorsten Schwarz, Karin Müller, and Rainer Stiefelhagen. 2024. Chartformer: A large vision language model for converting chart images into tactile accessible svgs. In *International Conference on Computers Helping People with Special Needs*, pages 299–305. Springer.

Jonas Pfeiffer, Gregor Geigle, Aishwarya Kamath, Jan-Martin O Steitz, Stefan Roth, Ivan Vulić, and Iryna Gurevych. 2021. xgqa: Cross-lingual visual question answering. *arXiv preprint arXiv:2109.06082*.

Chen Qiu, Dan Oneata, Emanuele Bugliarello, Stella Frank, and Desmond Elliott. 2022. Multilingual multimodal learning with machine translated text. *arXiv preprint arXiv:2210.13134*.

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

David Romero, Chenyang Lyu, Haryo Akbarianto Wibowo, Teresa Lynn, Injy Hamed, Aditya Nanda Kishore, Aishik Mandal, Alina Dragonetti, Artem Abzaliev, Atnafu Lambebo Tonja, and 1 others. 2024. Cvqa: Culturally-diverse multilingual visual question answering benchmark. *arXiv preprint arXiv:2406.05967*.

Andreas Steiner, André Susano Pinto, Michael Tschannen, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, and 1 others. 2024. Paligemma 2: A family of versatile vlms for transfer. *arXiv preprint arXiv:2412.03555*.

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

Jingqun Tang, Qi Liu, Yongjie Ye, Jinghui Lu, Shu Wei, Chunhui Lin, Wanqing Li, Mohamad Fitri Faiz Bin Mahmood, Hao Feng, Zhen Zhao, and 1 others. 2024. Mtvqa: Benchmarking multilingual text-centric visual question answering. *arXiv preprint arXiv:2405.11985*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, and 1 others. 2024a. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024b. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Ashmal Vayani, Dinura Dissanayake, Hasindri Watawana, Noor Ahsan, Nevasini Sasikumar, Omkar Thawakar, Henok Biadglign Ademtew, Yahya Hmaiti, Amandeep Kumar, Kartik Kuckreja, and 1 others. 2024. All languages matter: Evaluating lmms on culturally diverse 100 languages. *arXiv preprint arXiv:2411.16508*.

Hongyu Wang, Jiayu Xu, Senwei Xie, Ruiping Wang, Jialin Li, Zhaojie Xie, Bin Zhang, Chuyan Xiong, and Xilin Chen. 2024a. M4u: Evaluating multilingual understanding and reasoning for large multimodal models. *arXiv preprint arXiv:2405.15638*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024b. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, and 1 others. 2024c. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. *Advances in Neural Information Processing Systems*, 37:113569–113697.

Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, Zhenda Xie, Yu Wu, Kai Hu, Jiawei Wang, Yaofeng Sun, Yukun Li, Yishi Piao, Kang Guan, Aixin Liu, and 8 others. 2024. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *Preprint*, arXiv:2412.10302.

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

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

Weihao Xuan, Rui Yang, Heli Qi, Qingcheng Zeng, Yunze Xiao, Yun Xing, Junjue Wang, Huitao Li, Xin Li, Kunyu Yu, and 1 others. 2025. Mmlu-prox: A multilingual benchmark for advanced large language model evaluation. *arXiv preprint arXiv:2503.10497*.

Cheng Yang, Chufan Shi, Yaxin Liu, Bo Shui, Junjie Wang, Mohan Jing, Linran Xu, Xinyu Zhu, Siheng Li, Yuxiang Zhang, and 1 others. 2024. Chartmimic: Evaluating lmm's cross-modal reasoning capability via chart-to-code generation. *arXiv preprint arXiv:2406.09961*.

Xinmiao Yu, Xiaocheng Feng, Yun Li, Minghui Liao, Ya-Qi Yu, Xiachong Feng, Weihong Zhong, Ruihan Chen, Mengkang Hu, Jihao Wu, and 1 others. 2025. Cross-lingual text-rich visual comprehension: An information theory perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 9680–9688.

Xiang Yue, Yueqi Song, Akari Asai, Seungone Kim, Jean de Dieu Nyandwi, Simran Khanuja, Anjali Kantharuban, Lintang Sutawika, Sathyanarayanan Ramamoorthy, and Graham Neubig. 2024. Pangea: A fully open multilingual multimodal llm for 39 languages. In *The Thirteenth International Conference on Learning Representations*.

Wenxuan Zhang, Mahani Aljunied, Chang Gao, Yew Ken Chia, and Lidong Bing. 2023. M3exam: A multilingual, multimodal, multilevel benchmark for examining large language models. *Advances in Neural Information Processing Systems*, 36:5484–5505.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, and 1 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*.

A Detailed Dataset Statistics

A.1 PolyChartQA

A.1.1 Data Statistics by Language and Chart Type

We show the detailed statistics of PolyChartQA in Tables 5 and 6, including per-language and per-chart-type breakdowns for both images and QA pairs.

A.1.2 Question and Answer Length Statistics

We report statistics of question and answer lengths across all ten languages in PolyChartQA, using token counts computed with the GPT-4o tokenizer. The distribution for each language, aggregated over training and test splits, is illustrated in Figure 5. These results highlight significant variation in textual length, which reflects both linguistic and orthographic diversity across languages.

A.1.3 Distribution of Images and Questions by Language

We further examine the distribution of images and questions in each language. Figure 6 presents a t-SNE visualization of CLIP image embeddings, while Figure 7 visualizes CLIP text embeddings of questions. In both cases, each subplot corresponds to a specific language. All points are uniformly colored to emphasize intra-language distribution rather than inter-category variation. These visualizations reveal the diversity and clustering patterns present in the multilingual data.

A.1.4 Distribution of Images, Questions, JSON, and Code for English Data

We also provide a detailed analysis of the English subset, which serves as the seed data for PolyChartQA. Figure 8 shows t-SNE visualizations of image and question embeddings, with points colored by chart type to reveal clustering based on visual and semantic chart characteristics. Figure 9 presents t-SNE plots of embeddings from the JSON data underlying the charts and the Python code used to generate them, again colored by chart type. These analyses illustrate the extent to which chart types can be distinguished within visual, textual, and structural representations.

A.2 PolyChart-Instruct

A.2.1 Data Statistics by Language and Chart Type

We show the detailed statistics of in PolyChart-Instruct in Tables 7 and 8, including per-language and per-chart-type breakdowns for both images and QA pairs.

B Human Evaluation Details

B.1 Information of Human Annotators

We conducted a rigorous human evaluation to measure the quality of multilingual chart images and theirs question-answering pairs in PolyChartQA

Table 5: Detailed statistics of Image counts per chart type across all languages in POLYCHARTQA.

Chart Type	EN	AR	BN	ES	FR	HI	JA	RU	UR	ZH	Total
3d-bar	40	31	27	35	35	30	26	30	30	26	**310**
area	106	79	76	84	78	86	61	65	68	63	**766**
bar	600	447	507	505	471	547	409	477	514	393	**4870**
box	171	144	155	148	144	153	131	132	153	134	**1465**
bubble	81	32	39	38	38	40	33	35	37	35	**408**
candlestick	86	62	67	74	62	70	50	56	61	56	**644**
funnel	211	148	155	158	154	165	121	142	137	117	**1508**
heatmap	183	133	149	149	153	160	120	134	153	125	**1459**
histogram	219	167	177	180	187	182	141	162	181	137	**1733**
line	600	491	500	551	521	539	436	516	509	402	**5065**
multi-axes	77	49	53	52	58	55	42	48	58	45	**537**
pie	190	133	148	150	148	162	120	130	146	93	**1420**
radar	42	23	25	26	24	29	27	24	26	21	**267**
rings	123	80	83	91	95	92	72	66	85	76	**863**
rose	84	46	58	53	61	64	36	44	54	34	**534**
treemap	104	74	78	85	75	78	68	63	72	60	**757**
Total	**2917**	**2139**	**2297**	**2379**	**2304**	**2452**	**1893**	**2124**	**2284**	**1817**	**22606**

Table 6: Detailed statistics of Question-Answer (QA) pair counts per chart type across all languages in POLYCHARTQA

Chart Type	EN	AR	BN	ES	FR	HI	JA	RU	UR	ZH	Total
3d-bar	40	31	27	35	35	30	26	30	30	26	**310**
area	107	80	77	85	79	87	62	66	69	64	**776**
bar	696	592	670	669	627	733	535	638	685	517	**6362**
box	171	144	155	148	144	153	131	132	153	134	**1465**
bubble	81	32	39	38	38	40	33	35	37	35	**408**
candlestick	86	62	67	74	62	70	50	56	61	56	**644**
funnel	211	148	155	158	154	165	121	142	137	117	**1508**
heatmap	183	133	149	149	153	160	120	134	153	125	**1459**
histogram	219	167	177	180	187	182	141	162	181	137	**1733**
line	646	689	718	794	739	770	602	734	720	551	**6963**
multi-axes	77	49	53	52	58	55	42	48	58	45	**537**
pie	210	146	164	165	163	178	129	145	159	106	**1565**
radar	42	23	25	26	24	29	27	24	26	21	**267**
rings	123	80	83	91	95	92	72	66	85	76	**863**
rose	84	46	58	53	61	64	36	44	54	34	**534**
treemap	104	74	78	85	75	78	68	63	72	60	**757**
Total	**3080**	**2496**	**2695**	**2802**	**2694**	**2886**	**2195**	**2519**	**2680**	**2104**	**26151**

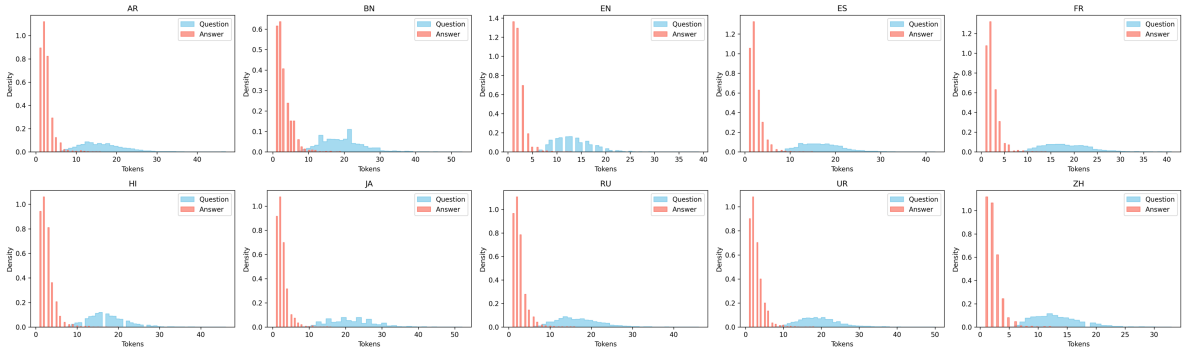


Figure 5: Question and answer length statistics in PolyChartQA.

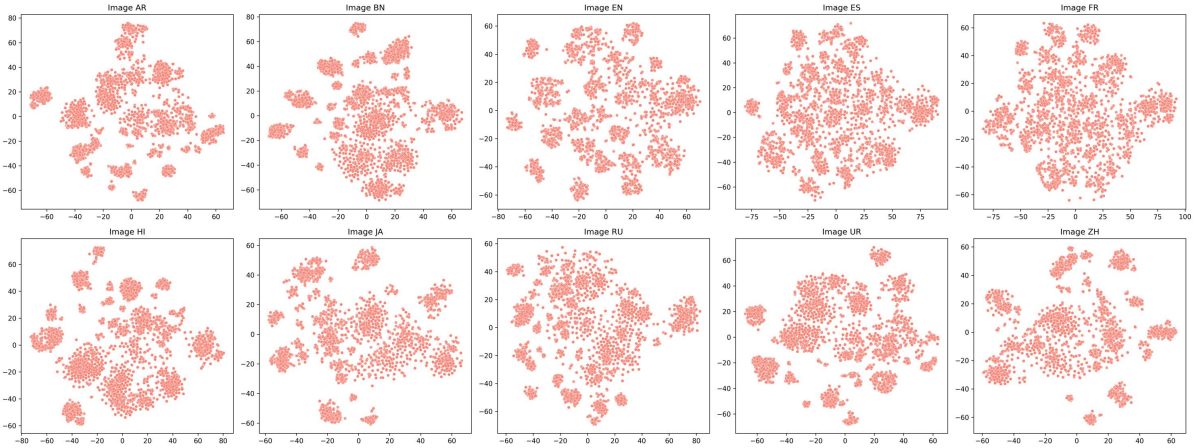


Figure 6: Distribution of images in PolyChartQA by language.

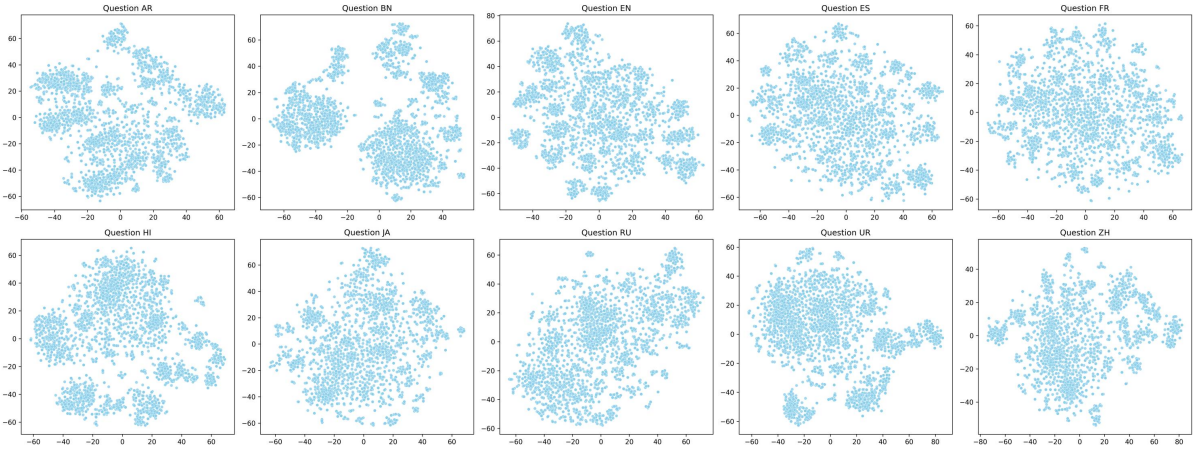


Figure 7: Distribution of questions in PolyChartQA by language.

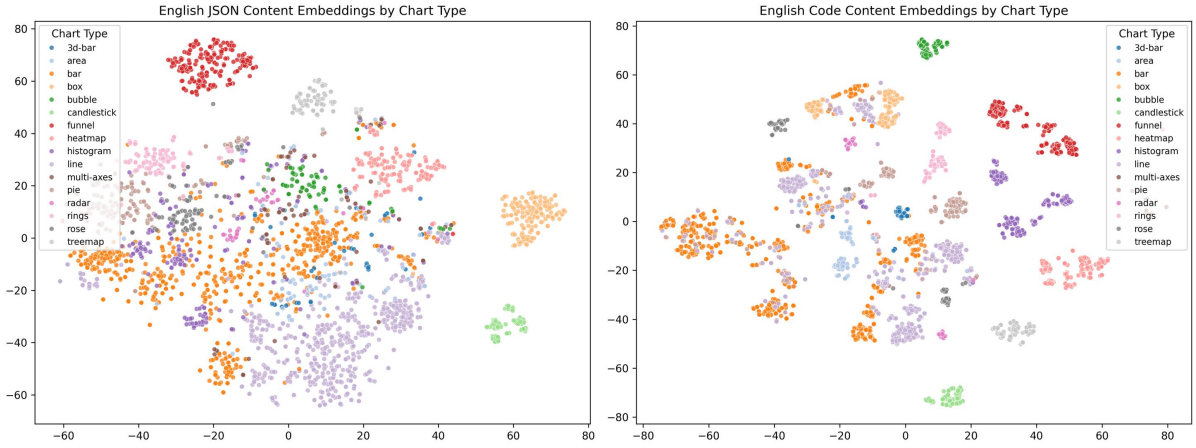


Figure 8: Distribution of images and questions in English by chart type in PolyChartQA.

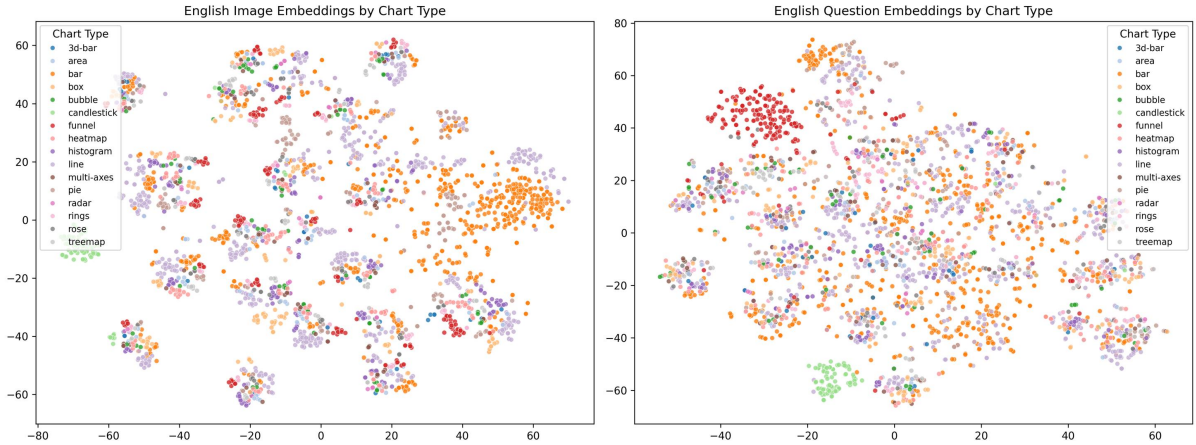


Figure 9: Distribution of JSON data and code in English by chart type in PolyChartQA.

Table 7: Detailed statistics of Image counts per chart type across all languages in PolyChart-Instruct.

Chart Type	AR	BN	EN	ES	FR	HI	JA	RU	UR	ZH	Total
3d-bar	4	4	4	2	3	3	3	2	4	4	**33**
area	1	1	1	1	1	1	1	1	1	1	**10**
bar	7834	7978	8876	7726	7878	8049	7883	7955	7804	8000	**79983**
box	50	47	57	47	48	46	46	48	49	50	**488**
candlestick	231	224	267	226	240	244	222	223	223	231	**2331**
funnel	103	107	118	101	96	107	102	106	100	102	**1042**
gantt	110	101	143	122	119	122	114	117	99	114	**1161**
heatmap	154	160	218	162	155	167	168	169	174	153	**1680**
line	3281	3383	3937	3220	3281	3374	3294	3374	3348	3340	**33832**
other	13	17	17	14	16	17	16	15	14	13	**152**
pie	630	629	781	602	593	645	632	643	631	630	**6416**
radar	184	176	203	166	165	185	165	177	167	173	**1761**
rings	66	68	88	67	69	68	68	69	68	70	**699**
scatter	186	193	222	182	185	185	188	200	187	196	**1924**
Total	**12847**	**13088**	**14932**	**12638**	**12849**	**13213**	**12902**	**13099**	**12869**	**13078**	**131515**

Table 8: Detailed statistics of QA pair counts per chart type across all languages in PolyChart-Instruct.

Chart Type	AR	BN	EN	ES	FR	HI	JA	RU	UR	ZH	Total
3d-bar	41	41	41	21	30	31	30	20	41	41	**317**
area	1	1	1	1	1	1	1	1	1	1	**10**
bar	33161	33764	38339	32794	33463	33940	33385	33626	32962	33998	**339432**
box	510	479	580	478	491	467	468	488	500	510	**4971**
candlestick	2279	2209	2639	2231	2369	2409	2190	2208	2201	2288	**23023**
funnel	1055	1097	1202	1042	982	1096	1055	1086	1009	1044	**10724**
gantt	1098	1008	1428	1218	1188	1219	1139	1169	989	1138	**11592**
heatmap	1547	1609	2190	1629	1558	1679	1688	1698	1750	1539	**16887**
line	25110	25942	30793	24645	25110	26013	25196	25998	25539	25763	**260109**
other	98	129	129	91	128	129	119	109	98	115	**1145**
pie	3833	3779	4901	3617	3615	3947	3863	3909	3806	3856	**39126**
radar	1845	1766	2042	1660	1662	1860	1663	1774	1671	1745	**17688**
rings	669	688	893	680	700	688	691	700	688	711	**7108**
scatter	1857	1933	2221	1824	1856	1857	1885	1997	1880	1955	**19265**
Total	**73104**	**74445**	**87399**	**71931**	**73153**	**75336**	**73373**	**74783**	**73135**	**74704**	**751363**

CHARTQA. All annotators are either native speakers with over 15 years of experience in the target language or individuals holding a bachelor's degree and official certification in the corresponding language. We recruit two annotators for each language.

B.2 Annotation Process

All annotations were collected via crowdsourcing. Annotators reviewed HTML-rendered charts and questions, and recorded their responses in structured Excel spreadsheets. Figure 11 shows an example of the custom annotation interface designed for this task, enabling annotators to efficiently compare original and translated chart images as well as their corresponding question-answer pairs.

C More Evaluation and Training Details

C.1 Source Dataset Licenses

We use three existing chart QA datasets as part of our data construction pipeline. CHARTQA is released under the GPL-3.0 license[1], CHARTX under the CC-BY-4.0 license[2], and CHARTLLAMA under the MIT license[3]. All datasets are publicly available via HuggingFace and used in accordance with their respective licenses.

[1] https://huggingface.co/datasets/ahmed-masry/ChartQA

[2] https://huggingface.co/datasets/U4R/ChartX

[3] https://huggingface.co/datasets/listen2you002/ChartLlama-Dataset

C.2 Implementation Details

For METEOR metric, we use its official code from huggingface[4].

C.3 Data Construction for Alignment Stage

The alignment stage (i.e., fine-tuning stage 1) is trained on four data sources, totaling approximately 850K samples:

1. **PolyChart-Instruct.** We extract image–JSON pairs from POLYCHART-INSTRUCT, yielding approximately 131K instances.

2. **MTVQA.** We incorporate the full training split of MTVQA (Tang et al., 2024), which contains 21K chart–QA pairs.

3. **Pangea.** We include 300K OCR data samples from the Pangea-OCR dataset (Yue et al., 2024).

4. **SMPQA-Reconstructed.** Following Geigle et al. (2025), we adapt SMPQA to our 10-language setting by reconstructing 410K synthetic chart-OCR training examples.

D More Experiments

D.1 Few-shot Results for Qwen2.5-VL

Table 9 presents few-shot performance on Qwen2.5-VL models under 0, 2, 4, and 8-shot settings. We observe that few-shot prompting does

[4] https://huggingface.co/spaces/evaluate-metric/meteor

Table 9: Few-shot performance of Qwen2.5-VL 3B and 7B models. Bold values denote the best performance.

Model Size	Shots	Avg.(w/ EN)	Avg.(w/o EN)
Qwen2.5-VL-3B	0	**0.537**	**0.518**
	2	0.505	0.493
	4	0.507	0.490
	8	0.515	0.499
Qwen2.5-VL-7B	0	0.538	0.530
	2	0.538	0.526
	4	0.559	0.545
	8	**0.560**	**0.548**

not reliably improve multilingual performance, suggesting that few-shot prompting alone is insufficient to address the multilingual transfer gap in current LVLMs.

D.2 Ablation on Training LoRA Rank

We conduct an ablation study to evaluate the impact of different LoRA ranks (32, 64, 128) on multilingual chart QA performance for Qwen2.5-VL models. As shown in Table 10, a higher rank generally leads to better performance, with rank 128 yielding the best average accuracy for both the 3B and 7B models. These results suggest that increased parameter capacity during low-rank adaptation improves multilingual generalization.

D.3 Ablation on English Data Percentage

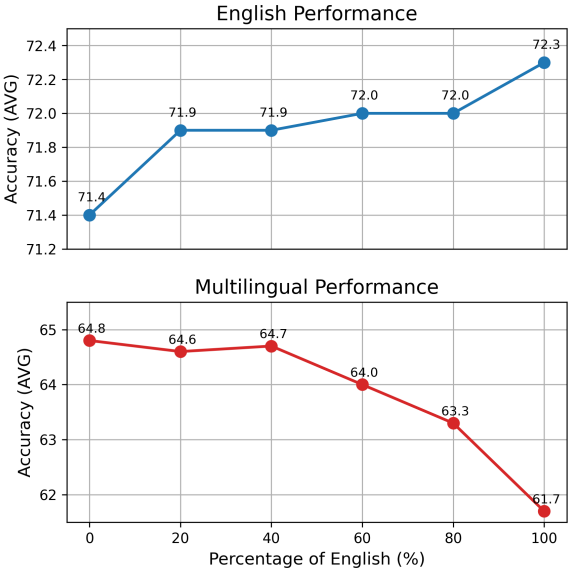


Figure 10: Performance on PolyChartQA varies with the proportion of English data.

To investigate the impact of English data proportion in multilingual fine-tuning, we conduct an ablation study by varying the ratio of English samples from 0% to 100% while keeping the total dataset size fixed at 70K QA pairs. The remaining proportion (i.e., non-English data) is evenly distributed across the other nine languages to ensure balanced multilingual representation. As shown in Figure 10, increasing the English ratio slightly improves English performance but consistently degrades accuracy on non-English languages, especially low-resource ones like Urdu, Hindi, and Bengali. These results highlight the trade-off between English performance and cross-lingual generalization, emphasizing the importance of balancing multilingual data in training. Full results are shown in Table 11.

D.4 Ablation on Monolingual Training

To analyze the effects of single-language supervision, we conduct a monolingual fine-tuning ablation using Qwen2.5-VL-7B, where each model is trained exclusively on data from one language. For fairness, we sample a fixed 70K QA pairs for each training setting. The "Balanced" variant divides the 70K budget equally across all ten languages. As shown in Table 12, models generally perform best in their own training language (diagonal), confirming strong in-domain alignment. Interestingly, some languages such as Arabic and Hindi also yield competitive performance across others, demonstrating potential for cross-lingual transfer. However, the fully balanced multilingual model still achieves strong average results, suggesting that multilingual supervision provides more consistent generalization than monolingual specialization.

D.5 Ablation Results on Qwen-2.5-VL Fine-tuning

Table 13 and Table 14 present the full ablation results of Qwen2.5-VL-3B and 7B, respectively. Across both model sizes, we observe consistent patterns: (i) fine-tuning alone provides substantial gains over the baseline, and (ii) incorporating an additional alignment stage further improves performance. Notably, the configuration where the vision encoder is unfrozen during alignment but frozen during instruction tuning achieves the highest accuracy in both models (63.6 for 3B, 68.0 for 7B). These results confirm that gradual visual adaptation followed by stabilization is a robust strategy for enhancing multilingual chart understanding across

Table 10: LoRA rank ablation study for Qwen2.5-VL 3B and 7B models. Bold values denote the best performance.

Model	Rank	AR	BN	EN	ES	FR	HI	JA	RU	UR	ZH	Avg. (w/ EN)	Avg. (w/o EN)
Qwen2.5-VL-3B	128	**59.0**	**54.0**	**68.2**	**65.4**	**66.1**	**56.8**	**63.1**	**64.9**	**49.8**	**64.1**	**61.1**	**60.2**
	64	58.3	52.6	67.8	65.1	**66.1**	55.2	61.1	64.3	48.7	63.4	60.3	59.3
	32	58.3	50.5	67.7	64.1	65.1	53.8	59.8	63.4	47.7	62.8	59.3	58.2
Qwen2.5-VL-7B	128	**65.5**	**60.9**	**73.1**	**70.0**	**71.1**	**64.9**	67.7	68.5	58.6	**68.5**	**66.9**	**66.1**
	64	65.0	60.7	72.8	**70.0**	70.9	64.4	**67.8**	**69.2**	58.5	**68.5**	66.8	66.0
	32	64.7	60.5	72.6	69.9	70.3	64.1	67.7	68.7	**59.0**	**68.5**	66.6	65.8

Table 11: Ablation study on the ratio of English data in the fine-tuning set. Bold values denote the best performance.

English Ratio (%)	EN	ZH	FR	ES	RU	JA	AR	HI	UR	BN	Avg. (w/ EN)	Avg. (w/o EN)
0	71.4	68.0	70.2	69.5	**68.9**	66.2	**63.4**	**62.4**	**56.6**	**58.8**	**65.5**	**64.8**
20	71.9	**68.3**	69.5	69.4	67.9	**67.2**	63.1	62.1	56.2	**58.8**	65.4	64.6
40	71.9	67.9	**70.5**	69.7	68.1	66.5	63.1	62.3	56.3	58.6	**65.5**	64.7
60	72.0	67.3	69.9	**69.8**	67.6	66.1	62.2	61.0	54.9	58.5	64.9	64.0
80	72.0	66.5	69.2	69.4	67.2	64.6	61.2	60.6	54.3	57.5	64.3	63.3
100	**72.3**	63.5	69.7	69.4	67.1	59.6	61.5	58.6	51.1	54.8	62.9	61.7

Table 12: Ablation study on single-language fine-tuning for Qwen2.5-VL-7B. The diagonal (in-domain) and lower triangle (cross-lingual transfer) are highlighted to show performance patterns. The best result in each evaluation column is in **bold**.

Training language	AR	BN	EN	ES	FR	HI	JA	RU	UR	ZH	Avg. (w/ EN)	Avg. (w/o EN)
AR	**66.0**	59.7	**72.2**	69.4	**70.5**	62.5	65.2	**69.6**	54.0	68.0	**65.7**	**64.9**
BN	60.4	**60.6**	69.0	67.4	66.3	62.0	65.0	65.7	53.5	66.2	63.6	62.9
EN	61.9	55.6	**72.2**	69.4	69.7	59.2	59.4	67.3	51.3	63.8	63.2	62.0
ES	60.3	55.5	70.2	**69.7**	69.8	59.1	62.5	67.4	51.7	64.4	63.1	62.2
FR	61.9	57.4	**72.2**	69.5	70.2	60.5	64.4	68.7	52.3	66.4	64.4	63.4
HI	63.1	59.8	72.0	69.4	**70.5**	**65.1**	63.9	69.4	55.4	67.2	**65.7**	64.8
JA	59.6	56.0	70.7	67.9	67.3	59.2	66.5	66.5	52.0	66.3	63.2	62.2
RU	61.9	57.3	72.0	**69.9**	**70.5**	60.3	65.3	67.9	52.6	67.3	64.5	63.5
UR	62.3	58.0	70.9	68.8	68.8	62.4	64.6	68.6	**60.1**	66.8	65.2	64.4
ZH	61.3	56.4	71.2	69.3	69.7	59.6	66.4	67.6	51.1	**68.5**	64.1	63.1
Balanced	62.9	58.5	71.8	69.5	70.3	62.9	**66.9**	68.7	56.3	68.1	65.6	64.8

different model scales.

Table 13: Performance of different training strategies on Qwen2.5-VL-3B across various languages. ❄ and 🔥 indicate that the vision encoder is frozen or unfrozen, respectively, during each stage. ✗ denotes that the stage is skipped. Bold values denote the best performance.

Training Strategy	Stage1	Stage2	EN	ZH	FR	ES	RU	JA	AR	UR	HI	BN	Avg. (w EN)	Avg. (w/o EN)
Baseline	✗	✗	67.4	59.6	61.8	62.5	58.0	48.8	51.4	37.2	45.7	43.0	53.7	51.8
SFT only	✗	❄	68.2	64.1	66.1	65.4	64.9	63.1	59.0	49.8	56.8	54.0	61.1	60.2
	✗	🔥	68.2	64.0	**66.3**	65.9	65.0	63.3	60.7	51.5	58.8	55.5	61.9	61.1
Align+SFT	❄	❄	68.8	64.2	66.1	**66.2**	64.3	62.7	61.3	53.4	57.9	53.5	61.9	60.9
	🔥	❄	69.0	**64.8**	65.5	**66.2**	65.2	**64.5**	**63.9**	**56.5**	**61.3**	**58.4**	**63.6**	**62.8**
	🔥	🔥	**69.3**	64.1	64.9	66.0	**65.5**	**64.5**	63.8	55.6	61.1	**58.4**	63.4	62.6

Table 14: Performance of different training strategies on Qwen2.5-VL-7B across various languages. ❄ and 🔥 indicate that the vision encoder is frozen or unfrozen, respectively, during each stage. ✗ denotes that the stage is skipped. Bold values denote the best performance.

Training Strategy	Stage1	Stage2	EN	ZH	FR	ES	RU	JA	AR	UR	HI	BN	Avg. (w EN)	Avg. (w/o EN)
Baseline	✗	✗	53.8	53.0	53.0	53.0	53.0	53.0	53.0	53.0	53.0	53.0	53.8	53.0
SFT only	✗	❄	73.1	68.5	71.1	70.0	68.5	67.7	65.5	58.6	64.9	60.9	66.9	66.1
	✗	🔥	72.6	68.8	70.6	70.0	**68.6**	67.9	65.1	60.0	65.2	61.6	67.0	66.2
Align+SFT	❄	❄	**73.6**	**69.6**	70.9	**70.8**	67.8	67.8	65.6	61.0	65.1	62.2	67.5	66.7
	🔥	❄	73.7	69.2	**71.3**	70.7	**68.0**	**68.0**	**66.1**	**62.7**	**66.6**	**62.9**	**68.0**	**67.2**
	🔥	🔥	73.5	69.1	70.4	70.2	68.2	68.1	65.3	59.6	64.4	62.1	67.1	66.3

Evaluation Guidelines

You will be provided with two chart images (one in English, one in the target language) and their corresponding Question-Answer (QA) pairs. Your task is to critically evaluate the target language materials based on the three dimensions below.

General Guidelines:

- **Reference:** Use the English materials as a reference for comparison.
- **Evaluation:** For each of the three dimensions, provide a score from 1 to 3.

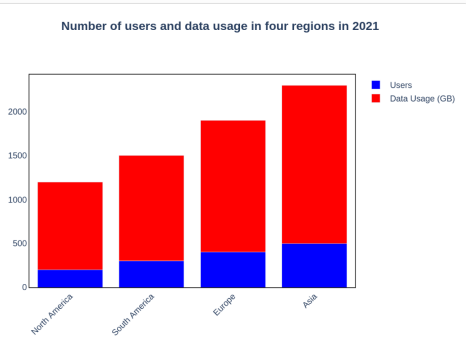
Evaluation Dimensions & Criteria:

Score	Description
Image Quality Assessment: Assess the visual quality of the target language chart. Evaluate its clarity, the legibility and correctness of all text and graphical elements, and its overall professional integrity.	
3	The image is clear, professional, and undistorted. All text and graphical elements are correctly displayed and legible. The chart type accurately reflects the data.
2	The chart has minor flaws, such as slight blurriness or minor display issues, but these do not significantly hinder comprehension.
1	The chart has major issues (e.g., distortion, illegible text, incorrect chart type) that hinder or prevent comprehension.
QA Correctness Assessment: Assess if the question is relevant to the chart and if the answer is factually correct and fully supported by the information presented in the target language chart.	
3	The question is relevant, and the answer is correct and fully supported by the chart data.
2	The QA pair has minor errors or ambiguities. The question might be slightly unclear, or the answer may have small inaccuracies.
1	The question is irrelevant to the chart, or the answer is factually incorrect or unsupported by the chart.
Translation Accuracy: Evaluate the quality of the image and QA translation from English to the target language. Assess its fidelity, semantic consistency, and natural fluency, and check if it conforms to the target language's idiomatic expressions. Crucially, determine if the translation introduces any bias, misinformation, or framing.	
3	The translation is accurate, fluent, and natural, conforming perfectly to the target language's conventions. It preserves the original meaning and key information without introducing any bias, misinformation, or framing.
2	The translation is mostly correct and preserves the core meaning, but has minor issues like awkward phrasing or does not feel fully idiomatic. It may subtly introduce minor bias or framing, but does not significantly mislead.
1	The translation has major errors, is semantically inconsistent, or is highly unnatural. Additionally, or as a primary issue, it introduces clear bias, misinformation, or framing that distorts the original message.

Evaluation Samples

Sample ID: bar_102_ar_001

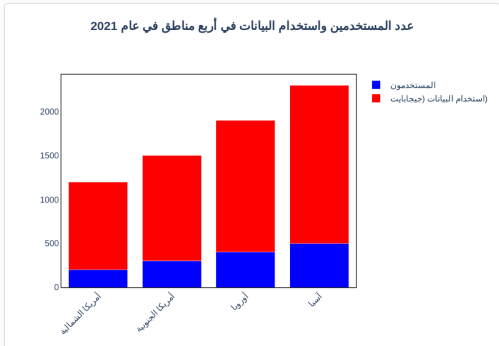
English Original



Q: How much data usage is reported in Asia?

A: 1800 GB

Translated Version



Q: ما هو حجم استخدام البيانات المسجل في آسيا؟

A: 1800 جيجابايت

Figure 11: Human evaluation interface. Annotators review chart images and QA pairs in both source and target languages, providing quality ratings for image quality, QA correctness and translation accuracy.

E Full Prompt Templates Used in Our Study

In this appendix, we present all prompt templates used throughout our PolyChartQA data pipeline. This includes the pipeline prompts for data cleaning, generation, translation, and consistency checking.

Stage 1 Prompt for Question-Answer Pair Rewriting

You are a data processing expert specializing in refining chart Question-Answering pairs for automated evaluation.
Your goal is to process provided Question-Answer examples, classifying them (`KEPT`, `MODIFIED`, `DELETE`) and potentially shortening the `label` (answer) to a concise format suitable for exact match (or numerical match with tolerance) evaluation.

CORE INSTRUCTION: Assess the provided `label` in the context of the `query`. You MUST base the `new_label` strictly on information present in the original `label`. **Do NOT generate new information or answers.**

Input:

1. `query`: The question asked about a chart.
2. `label`: The original answer.

Task Steps (Follow Strictly):

1. **Assess Query Suitability (DELETE):**
   If the `query` requires an answer that cannot be concise (e.g., trend, explanation, subjective, or complex comparison), set `action: "DELETE"`, `new_label: ""`, and stop.
2. **Assess Label Conciseness (KEPT):**
   If the original `label` is already concise (single number, name, yes/no, short list, or "Unanswerable"), set `action: "KEPT"`, `new_label: label` (exact copy), and stop.
3. **Perform Modification (MODIFIED):**
   If the query is suitable and the label is verbose, set `action: "MODIFIED"`, extract ONLY the core factual answer(s), format concisely (list, units, standardize "Data not available" as "Unanswerable"), and set as `new_label`.

Final Output Format:
Respond ONLY with the following JSON object (no other text):

```
{
  "action": "KEPT" | "MODIFIED" | "DELETE", "new_label": "string"
}
```

If action is DELETE, new_label must be ""; if KEPT, new_label is identical to the original label; if MODIFIED, new_label is your concise rewrite. Now, process the following input:

```
{ "query": "{query}", "label": "{label}" }
```

Stage 1 Prompt for Question-Answer Pair Rating

You are an expert evaluator for chart question-answering pairs.
Your task is to assess the quality and correctness of the provided **Answer** in response to the **Question**, based *solely* on the information presented in the accompanying chart image. Assign a rating from 1 to 5 based on the criteria below.

Do not use any external knowledge or make assumptions beyond what is visually represented or directly calculable from the chart.

Rating Scale and Criteria:

- **5: Excellent / Fully Correct**
  The answer is completely accurate according to the chart data; directly and fully addresses the question; all information is visible or calculable from the chart; no ambiguities or unsupported inferences.
- **4: Good / Mostly Correct**
  Substantially correct, with only very minor inaccuracies or omissions; main point addressed; clearly derived from the chart.
- **3: Fair / Partially Correct**
  Contains both correct and incorrect elements, or answers the wrong question, or relies on inferences not explicitly supported; addresses the question only partially or inaccurately.
- **2: Poor / Mostly Incorrect**
  Contains significant errors contradicted by the chart; fundamentally misunderstands the chart/question; core claim is wrong according to the chart.
- **1: Very Poor / Completely Incorrect or Irrelevant**
  Entirely false or irrelevant to the chart/question; no connection between answer and the visual evidence.

Input Context (User Prompt):

1. Chart Image
2. Chart Question
3. Proposed Answer

Output Format:
Respond ONLY with a valid JSON object containing:

```
{
  "rating": <integer 1-5>,
  "reason": "<brief justification, referencing specific chart elements or data points where possible>"
}
```

Example Output (Score 5):

```
{
  "rating": 5,
  "reason": "The answer accurately states the value for 'Q3 Revenue' is \$1.2M, which..."
}
```

Example Output (Score 3):

```
{
  "rating": 3,
  "reason": "The answer correctly identifies 'Product A' as having the highest value, but..."
}
```

Example Output (Score 1):

```
{
  "rating": 1,
  "reason": "The answer discusses stock market trends, which are completely absent from the provided..."
}
```

Now, evaluate the specific image, question, and answer provided in the user prompt based on the 1-5 scale. **Respond ONLY with the JSON object.**

Stage 2 Prompt for JSON and Code Extraction

You MUST act as an expert Python data visualization assistant. Your primary objective is to meticulously analyze a given chart image, extract its data and text into a structured JSON format suitable for translation, and then generate a robust Python script using Plotly that accurately recreates the chart **solely** from that JSON data. The generated script must preserve the original data order and handle multilingual text input correctly, in addition to proactively addressing potential layout issues.
Input:

1. `<image_description>`: A reference to, or the content of, the input chart image file.
2. `<image_filename_base>`: The base filename string for the input image (e.g., "my_chart"). This base name is crucial for naming the JSON file read by the script and the output PNG image.

Your Tasks (Execute Sequentially):

1. **Analyze Image and Generate JSON Data Structure:**
    - Identify chart type and store as `chart_type` if useful.
    - Extract all data series and categories (order must match original visual presentation). Store as `chart_data`.
    - Extract **all** visible text elements into a `texts` dictionary, preserving original English, capitalization, and line breaks (`<br>`). If an element is missing, set its value to `null`.
    - Extract primary colors as hex codes in a `colors` list, aligned with data series order.
    - Final JSON contains `chart_data`, `texts`, `colors`, and optionally `chart_type`.
2. **Generate Robust Python Plotly Code:**
    - **Data source:** The script must read only from `<filename>.json` and use the unpacked JSON for all chart content and styling. Absolutely no hardcoded data or text.
    - Use Plotly (`plotly.graph_objects`) to recreate the chart. Iterate through JSON data in order; apply colors and texts per JSON content.
    - Combine titles/subtitles and source/note using HTML as specified.
    - **Multilingual/Unicode support:** Code must be language-agnostic, display provided strings as-is, and handle non-Latin scripts without logic changes.
    - **Layout:** Prevent clipping/overlap with careful margins, anchors, and text placement. Font must be Arial.
    - Output PNG as `<filename>.png`, with `scale=2`.
    - Clean code: no extra installs, no function definitions, no unnecessary comments, only minimal print.

Output Format:
Return the output in **exactly** two code blocks:

- A single JSON code block containing the full JSON object.
- A single Python code block containing the full script.

`Here is the filename <FILENAME> and the chart image.`

Stage 2 Prompt for Visual Consistency Check and Visual Flaw Detection

You are an expert visual comparison and chart quality evaluator. Your task is to assess two chart images (**Original**, **Rendered**) based on two criteria: Semantic Consistency and Visual Flaws.
Input:

1. Original Chart Image
2. Rendered Chart Image (generated from code based on the original)

Task 1: Evaluate Semantic Consistency (Rating 1–5)
Assess if the Rendered Image accurately represents the *same core data and key information* as the Original Image. Focus on:

- **Data Values & Proportions**: Are numerical values (bars, points, slices) substantially the same? Do relative proportions match?
- **Categories & Series**: Do labels, axes, and legend entries match the original data structure and order?
- **Text Content**: Are Title, Axis Titles, Legend Labels, and other key text elements semantically identical or extremely close to the original?
- **Color Hue Consistency**: While exact shades may differ, do the primary colors for data series maintain the same *hue category* (e.g., reds stay red/orange, blues stay blue/cyan, greens stay green)? A swap from red to blue is a major inconsistency.
- **Overall Message/Trend**: Does the rendered chart convey the same main insight or pattern?

IGNORE minor stylistic differences (fonts, gridlines, spacing) UNLESS they hinder interpretation or violate the checks above.
Rating Scale (1–5):

- 5: **Highly Consistent**: Near-perfect semantic match in data, text, color hues, and overall message. Only negligible, non-misleading differences.
- 4: **Mostly Consistent**: Core data, text, and message are accurate. Minor data inaccuracies, text variations, or color shade differences (hue preserved), but interpretation unchanged.
- 3: **Moderately Consistent**: Some aspects captured, but noticeable discrepancies. Key values may be inaccurate, important text differs, color hues mismatched, or message partially distorted.
- 2: **Poorly Consistent**: Significant data errors, trends misrepresented, text is wrong/misleading, or color usage creates confusion. Fundamentally different interpretation.
- 1: **Inconsistent / Unrelated**: Completely different data, topic, or structure.

Task 2: Identify Visual Flaws (Yes/No)
Determine if the Rendered Image has significant visual flaws that impede understanding or indicate generation errors. Check for:

- **Severe Text Overlap**: Critical labels, titles, or data points overlapping illegibly.
- **Element Clipping**: Chart elements (data, labels, legends) cut off by boundaries.
- **Unreadable Text**: Text is too small, blurry, or has unsupported characters.
- **Data Obscurity**: Data points hidden behind other elements.
- **Empty/Malformed Chart**: Blank image, error messages, or not a meaningful chart.
- **Gross Layout Issues**: Elements positioned bizarrely, chart is nonsensical.

Answer **Yes** if any major flaws; **No** if not. Minor imperfections that do not hinder core interpretation = **No**.
Output Format:
Respond ONLY with a valid JSON object containing FOUR keys:

- `"similarity_rating"`: integer (1–5 based on Task 1)
- `"similarity_reason"`: string (brief explanation for the similarity rating)
- `"has_visual_flaws"`: boolean (true if significant flaws found, false otherwise)
- `"flaw_reason"`: string (brief explanation if flaws were found, otherwise "No significant flaws detected.")

Now, evaluate the Original and Rendered images based on BOTH tasks. Respond ONLY with the JSON object.

Stage 2 Prompt for QA Validity Check

You are an expert evaluator for chart question-answering pairs. Your task is to assess the quality and correctness of the provided 'Answer' in response to the 'Question', based **solely** on the information presented in the accompanying chart image. Assign a rating from 1 to 5 based on the criteria below.
Do not use any external knowledge or make assumptions beyond what is visually represented or directly calculable from the chart.
Rating Scale and Criteria:

- **5: Excellent / Fully Correct:**
  The answer is completely accurate according to the chart data; directly and fully addresses the question; all information is visible or directly calculable from the chart; no ambiguities or unsupported inferences.
- **4: Good / Mostly Correct:**
  Substantially correct; addresses main point; may contain very minor inaccuracies or omissions that do not significantly mislead.
- **3: Fair / Partially Correct:**
  Mix of correct and incorrect information; may extract data but fail to answer the question; may make unsupported inferences; partially or inaccurately addresses the question.
- **2: Poor / Mostly Incorrect:**
  Contains significant factual errors; fundamentally misunderstands chart or question; core claim is wrong based on chart evidence.
- **1: Very Poor / Completely Incorrect or Irrelevant:**
  Completely false or irrelevant; no connection between answer and the chart content.

Input Context:

1. Chart Image
2. Chart Question
3. Proposed Answer

Output Format:
You MUST respond ONLY with a valid JSON object containing two keys:

- `"rating"`: integer (1 to 5)
- `"reason"`: string (brief explanation for your assigned rating, referencing chart elements or data points where possible)

Example Output (Score 5):

```
{
  "rating": 5,
  "reason": "The answer accurately states the value for 'Q3 Revenue' is $1.2M, which..."
}
```

Example Output (Score 3):

```
{
  "rating": 3,
  "reason": "The answer correctly identifies 'Product A' as having the highest value, but..."
}
```

Example Output (Score 1):

```
{
  "rating": 1,
  "reason": "The answer discusses stock market trends, which are completely absent from the provided..."
}
```

Now, evaluate the specific image, question, and answer provided in the user prompt based on the 1-5 scale. Respond ONLY with the JSON object.

Stage 3 Prompt for Translation (Back-Translation)

You are an expert linguist and JSON data localization specialist simulating a translation process. Your task is to translate a given JSON object representing chart data & its associated question-answer pairs from `{source_language_name}` (`{source_language_code}`) to `{target_language_name}` (`{target_language_code}`). You must intelligently identify and translate only the user-facing text while preserving the JSON structure and non-textual data precisely.

Input Data:
You will receive a JSON object containing two keys:

1. `chart_json_data`: The JSON object extracted from a chart (variable structure).
2. `qa_pairs_to_translate`: A list of dictionaries, each with "query" and "label" strings in `{source_language_code}`.

CRITICAL Instructions for Translation:

1. **Goal:** Produce a translated version of the input suitable for displaying the chart and Q&A in `{target_language_name}`.
2. **Translate `chart_json_data` Recursively:**
   - Traverse the entire structure (nested dicts/lists).
   - **ONLY translate string values** meant for user display in `{source_language_name}` (titles, axis labels, legend entries, annotations, etc.).
   - **DO NOT translate/modify:**
     - JSON keys
     - Numerical values (int/float)
     - Strings only of numbers (e.g., "2023", "1.5")
     - Strings only of numbers with "%" (e.g., "55.5%", "-10%")
     - Hex color codes (e.g., "#1f77b4")
     - URLs, file paths, system identifiers
     - Boolean strings ("true", "false")
     - Type keywords (e.g., "stacked_bar", "Arial", "auto"). If unsure, do NOT translate.
     - `null` values and empty strings.
   - Preserve units and symbols unless a direct, standard equivalent is always used in `{target_language_name}`.
   - **Output JSON MUST be identical in structure and data types to input. ONLY translatable string values change.**
3. **Translate `qa_pairs_to_translate`:**
   - Translate "query" and "label" for each item.
   - **Consistency:** Use the exact same translation for terms that appear in both the chart JSON and QA pairs.
4. **Translation Quality Requirements:**
   - **Accuracy & Fidelity:** Preserve factual meaning.
   - **Naturalness & Fluency:** Use grammatically correct, natural phrasing.
   - **Consistency:** Identical source terms = identical translation.
   - **Cultural Appropriateness:** Target-audience appropriate.
   - **Linguistic Integrity:** Correct grammar, syntax, style.
   - **Vocabulary Usage:** Accurate and context-appropriate.
   - **Non-Latin/BiDi Support:** Generate correct Unicode. Standard rendering will handle text direction.
   - **HTML tags:** Preserve tags like `<br>` in correct position.

Output Format:
You MUST respond ONLY with a single, valid JSON object containing:

- `translated_chart_json`: The processed chart JSON, structure identical to input, translations ONLY on user-facing text.
- `translated_qa_pairs`: List of translated QA pairs in the original order, each with:
  - `translated_query`
  - `translated_label`

Input Data to Process:

Stage 3 Prompt for Semantic Consistency Evaluation of English Content

You are an expert linguistic evaluator comparing two versions of content in {source_language_name} ({source_language_code}). One is the 'Original Content', and the other is the 'Back-Translated Content' (which was translated to another language and then back to {source_language_name}).
Your task is to evaluate the semantic equivalence between the Original and Back-Translated content based on the provided context, assigning ratings on a 1-5 scale. The required output format depends on the provided context.
Input Format (Provided in User Prompt):
You will receive a JSON object with three keys:

1. context: A string indicating the type of content: either "Chart JSON Texts" or "Question-Answer Pair".
2. original_content: The original content (either a JSON object for chart texts or a dict like {"query": "...", "label": "..." } for a QA pair) in {source_language_name}.
3. back_translated_content: The back-translated content (matching the structure of original_content) in {source_language_name}.

Evaluation Criteria and Rating Scale (1-5):

- **Focus:** Semantic meaning and preservation of key information. Does the back-translation mean the same thing as the original?
- **Ignore:** Minor grammatical variations, stylistic choices, or synonymous phrasing common in translation *unless* they significantly alter the meaning, introduce ambiguity, or omit/distort critical information.
- 5: **Excellent Equivalence** — Perfect semantic match; only stylistic or trivial differences.
- 4: **Good Equivalence** — Main meaning and most key info conveyed accurately; minor acceptable differences.
- 3: **Fair Equivalence** — General topic captured, but some important details, nuance, or accuracy lost.
- 2: **Poor Equivalence** — Significant errors; key info lost, distorted, or contradicted.
- 1: **No Equivalence / Unrelated** — Meaning is completely different, nonsensical, or unrelated.

CRITICAL: Output Format Based on Context:
A. If context is "Chart JSON Texts":

- Evaluate the **overall semantic equivalence** of the translatable text content in back_translated_content JSON vs. original_content JSON.
- Respond ONLY with a single, valid JSON object with TWO keys:
  - rating: integer (1-5, overall JSON equivalence)
  - reason: string (brief justification)

Example:

```
{
  "rating": 4,
  "reason": "Overall JSON text equivalence is good. Most titles and labels match semantically, though..."
}
```

B. If context is "Question-Answer Pair":

- Evaluate the Query and the Label (Answer) separately.
- Respond ONLY with a valid JSON object containing FOUR keys:
  - query_rating: integer (1-5, query equivalence)
  - query_reason: string (brief justification for query)
  - label_rating: integer (1-5, label/answer equivalence)
  - label_reason: string (brief justification for label)

Example:

```
{
  "query_rating": 5,
  "query_reason": "Back-translated query perfectly matches original meaning.",
  "label_rating": 5,
  "label_reason": "Back-translated label is identical to original."
}
```

Final Instruction: Analyze the original_content and back_translated_content based on context. Respond ONLY with the valid JSON object matching the required output format for that context.