FinChart-Bench: Benchmarking Financial Chart Comprehension in Vision-Language Models

Anonymous ACL submission

Abstract

Large vision-language models (LVLMs) have made significant progress in chart understanding. However, financial charts, characterized by complex temporal structures and domain-specific terminology, remain notably underexplored. We introduce **FinChart-Bench**, the first benchmark specifically focused on real-world financial charts. FinChart-Bench comprises 1,200 financial chart images collected from 2015 to 2024, each annotated with True/False (TF), Multiple Choice (MC), and Question Answering (QA) questions, totaling 7,016 questions. We conducted a comprehensive evaluation of 26 state-of-the-art LVLMs on FinChart-Bench. Our evaluation reveals critical insights: (1) the performance gap between open-source and closed-source models is narrowing, (2) performance degradation occurs in upgraded models within families, (3) many models struggle with instruction following, (4) both advanced models show significant limitations in spatial reasoning abilities, and (5) current LVLMs are not reliable enough to serve as automated evaluators. These findings highlight important limitations in current LVLM capabilities for financial chart understanding. The FinChart-Bench dataset is available at https://anonymous.4open.science/r/FinChart-Bench-34FB.

1 Introduction

Large vision-language models (LVLMs) have achieved remarkable breakthroughs in chart understanding tasks, demonstrating their ability to interpret complex visual data representations and answer questions about statistical trends, data relationships, and quantitative insights (Huang et al., 2024; Han et al., 2023; Xu et al., 2024). To evaluate and advance these capabilities, several benchmarks have been developed specifically for chart understanding, such as Chart-to-text (Kantharaj et al., 2022), ChartBench (Xu et al., 2023), and ChartX (Xia et al., 2024). However, despite these advances, one of the most critical application domains, finance remains notably underexplored. Financial charts encode a tremendous amount of valuable information about market behavior, company performance, and economic trends. Unlocking this information through LVLM has the potential to transform the financial industry, representing opportunities worth billions of dollars. However, financial charts present unique challenges, including complex temporal patterns, dense visual layouts, and specialized terminology. To enhance and evaluate LVLMs' ability to understand financial charts, we introduce a dedicated benchmark designed specifically for this domain.

The design of effective chart understanding benchmarks faces key challenges that have hindered progress in this area. First, current practice increasingly relies on LVLMs themselves to generate benchmark datasets and serve as automated evaluators during construction. However, we argue that current LVLMs are not yet reliable enough to serve as automated evaluators during benchmark construction, necessitating substantial human effort to ensure data quality and accuracy. Second, many existing benchmarks suffer from significant ambiguity in their evaluation design. Recent trends in benchmark design have leaned toward overcomplication, making systematic evaluation increasingly difficult. Many existing benchmarks adopt question-answering formats with multi-step reasoning processes, which introduce significant ambiguity (Xia et al., 2024; Xu et al., 2023; Han et al., 2023; Liu et al., 2023a; Meng et al., 2024). For example, if the ground truth is "140 Million," should variants such as "140M," "$140M," or "140000000" be considered correct? This kind of ambiguity, coupled with the lack of finance-specific datasets, creates a significant gap in our ability to evaluate and improve the performance of LVLMs on financial chart understanding tasks.

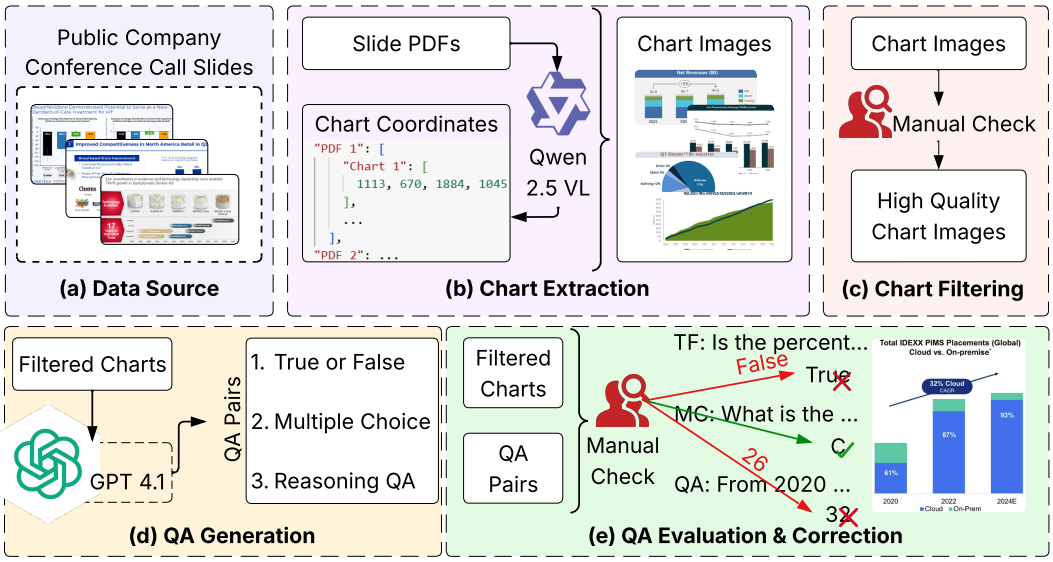


Figure 1: Overview of the FinChart-Bench generation pipeline. The process consists of five stages: (a) Data Source, (b) Chart Extraction, (c) Chart Filtering, (d) QA Generation, and (e) QA Evaluation & Correction. The pipeline utilizes Qwen 2.5 VL for automated chart extraction and GPT 4.1 for generating initial question-answer pairs across three distinct tasks. Furthermore, the workflow incorporates two rounds of rigorous manual human evaluation to filter chart images and verify answer accuracy, ensuring high data quality.

To address these challenges, we introduce **FinChart-Bench**, a new benchmark comprising 1,200 real-world financial chart images collected from the years 2015 to 2024. Each chart is annotated with three types of questions: True/False (TF), Multiple Choice (MC), and Question Answering (QA). Each chart contains one or two questions of each type, resulting in a total of 2,383 TF questions, 2,349 MC questions, and 2,284 QA questions (total of 7,016 questions). Unlike prior benchmarks that either do not involve human validation at all (Kahou et al., 2017; Kafle et al., 2018; Methani et al., 2020; Li and Tajbakhsh, 2023), or only perform a single pass of manual review on a subset of model-generated data (Xu et al., 2023; Xia et al., 2024; Lu et al., 2023; Wu et al., 2024a; Wang et al., 2024b; Li et al., 2024), our benchmark undergoes two rounds of rigorous human evaluation. In the first round, we manually filter and validate each chart image to ensure completeness, legibility, and relevance. In the second round, we thoroughly verify and refine each question–answer pair for accuracy and clarity. We conduct a comprehensive evaluation of over 20 leading LVLMs and uncover several findings: (1) The average performance gap between open-source and closed-source models is reducing, and closed-source models might reach their bottleneck on the chart understanding and reasoning task. (2) We found performance degradation in upgraded models within the same family. (3) Existing open-source LVLMs, especially those fine-tuned on charts, struggle with instruction following. (4) Both advanced open-source and closed-source models show significant limitations in "spatial ability". (5) Current LVLMs are not yet capable of serving as reliable judges when constructing a new benchmark. Our contributions are as follows:

- We introduce FinChart-Bench, the first benchmark focused on real-world financial charts. The entire benchmark are manually evaluated twice, addressing the lack of high-quality datasets in chart understanding.
- We design questions to have unambiguous, single-token answers, enabling straightforward and reproducible evaluation. Even simple metrics such as Exact Match (EM) are sufficient to reliably measure performance.
- We evaluate over 20 state-of-the-art LVLMs on FinChart-Bench and observe that current models still struggle with our benchmark. highlighting the limitations of LVLMs in financial chart understanding. These results highlight not only the limitations of LVLMs in financial chart understanding, but also validate our decision to rely on human judgment rather than automated evaluation throughout the benchmark development process.

Table 1: Benchmark comparison of existing chart datasets. To the best of our knowledge, FinChart-Bench is the first chart benchmark focused on the finance domain and the only one to undergo two rounds of manual evaluation.

Dataset	# Image	# Question	Quest Type	# Task	Metric	Domain	Human
FigureQA (Kahou et al., 2017)	180K	2.3M	Template	1	EM	Open-domain	No
DVQA (Kafle et al., 2018)	300K	348K	Template	1	EM	Open-domain	No
PlotQA (Methani et al., 2020)	224K	28.9M	Template	3	EM	Open-domain	No
ChartSFT (Meng et al., 2024)	39M	39M	Mixed	5	BLEU, GPT, etc.	Open-domain	No
ChartBench (Xu et al., 2023)	66.6K	600K	Mixed	5	GPT, Acc+, etc.	Open-domain	Subset
UniChart (Masry et al., 2023)	627K	7M	Mixed	3	Acc, GPT, etc.	Open-domain	Subset
SciCap (Hsu et al., 2021)	2.1M	2.1M	Free-form	1	BLEU, METEOR, etc.	Computer Science	No
M-Paper (Hu et al., 2024)	350K	702K	Free-form	3	BLEU, ROUGE, etc.	Deep Learning	No
ArXivCap (Li et al., 2024)	6.4M	100K	Free-form	1	BLEU, CIDEr, etc.	Open-domain	No
SciGraphQA (Li and Tajbakhsh, 2023)	295K	657K	Free-form	2	CIDEr, GPT, etc.	Computer Science	No
Chart-to-text (Kantharaj et al., 2022)	44K	44K	Free-form	1	BLEU, METEOR, etc.	Open-domain	Subset
ChartQA (Masry et al., 2022)	21.9K	32.7K	Free-form	4	Relaxed Acc, EM, etc.	Open-domain	Subset
ChartLLaMa (Han et al., 2023)	11K	160K	Free-form	7	GPT, EM, etc.	Open-domain	Subset
ChartX (Xia et al., 2024)	6K	6K	Free-form	7	GPT, EM, etc.	Open-domain	Subset
MMC-Bench (Liu et al., 2023a)	1K	2K	Free-form	9	GPT, Micro Acc, etc.	Open-domain	1 time
FinChart-Bench (Ours)	1.2K	7K	Free-form	3	EM	Finance	2 times

2 FinChart-Bench Construction

2.1 Data Processing Pipeline

As shown in Figure 1, FinChart-Bench was constructed through a five-stage pipeline designed to ensure the highest quality of both visual chart data and associated question-answering pairs. The following sections detail each stage of this process.

Data Source. We construct a comprehensive dataset of corporate presentation slides spanning the period 2015 to 2024, drawing from multiple sources including Bloomberg News and official corporate websites. Our sample encompasses a wide range of public executive presentations delivered in both firm-hosted settings (e.g., corporate-sponsored conferences) and third-party venues (e.g., investment bank-sponsored events). Notably, the majority of presentations in our dataset are externally hosted and do not constitute traditional roadshows. This diverse collection of real-world financial presentations provides an ideal foundation for constructing a comprehensive benchmark for financial chart understanding.

Chart Extraction. The second stage involved the automated extraction of chart images from our data source. The process began by converting each page of the PDFs into a high-resolution image. Each page image was then analyzed by the Qwen2.5-VL-7B-Instruct model (Bai et al., 2025), which was prompted to generate bounding box coordinates for any charts present. The used prompt is in Appendix C.1. Using the generated coordinates, each potential chart was programmatically cropped from its page image and saved for the subsequent stage. This process results in more than 130,000 charts.

Chart Filtering. The automated extraction process yielded numerous images, but the quality varied significantly. Therefore, this stage involved a manual filtering process to create a high-quality dataset. Our team evaluated each extracted image against a strict set of criteria, retaining only those that met every requirement: (1) The image must be a chart (e.g. bar, line, pie, etc.). (2) The chart must be completed, with no truncated or missing sections. All essential information, including axis labels, legends, and data, must be clearly legible. (3) The chart must be high-resolution, free of distracting background elements, watermarks, or significant compression artifacts. (4) The chart should not be overly simplistic nor excessively complex to the point of being indecipherable. We adopted a conservative approach, discarding any borderline or uncertain cases. This process yielded over 71,000 high-quality chart images, averaging around 7,000 images per year from 2015 to 2024. Detailed statistic is shown in Figure 2a. To minimize human labor in later stages, we manually selected 120 top-quality charts per year, resulting in a final dataset of 1,200 charts. Examples of accepted and rejected charts are shown in Figure 4 (Appendix).

Since the subsequent pipeline involves human evaluation and correction, to reduce human labor, we manually selected 120 of the highest-quality chart images from each year, resulting in a refined dataset of 1,200 charts. Examples of accepted and rejected charts are provided in Figure 4 in Appendix.

QA Generation. With the curated set of 1,200 charts, we proceeded to generate question-answer pairs using GPT-4.1 (Hurst et al., 2024). To ensure a diverse and comprehensive benchmark, we

targeted three distinct task types: True/False, Multiple Choice, and Reasoning QA. For each chart, we prompted the model to generate two QA pairs for each task type, using task-specific prompts detailed in Appendix C.2. Generating two pairs per task increases the balance and diversity of our benchmark, particularly for the True/False category, helped ensure a balanced distribution, achieving a nearly 50% ratio of True to False answers in the dataset.

A central design goal for FinChart-Bench was to create questions that require reasoning while yielding answers that are unambiguous and simple to evaluate. To this end, all answers were constrained to a single token. For True/False, the answers are strictly "True" or "False". For Multiple Choice, the answer is the correct letter ("A", "B", "C", or "D"). For Reasoning QA, which involves chart reasoning and calculations, we instructed the model to provide both the final numerical answer and the step-by-step reasoning used to derive it. However, only the single-number result is designated as the ground-truth answer, with the reasoning process stored separately as supplementary metadata. This stage concluded with the generation of 7,200 QA pairs (1,200 charts × 3 tasks × 2 pairs).

QA Evaluation & Correction. The final stage of our pipeline was a second manual review focusing on the 7,200 generated QA pairs. This verification step was critical for ensuring the accuracy and integrity of the benchmark. Each QA pair was individually assessed by a human according to the following protocol: (1) Correct QA pairs were left unchanged. (2) For pairs with a correct question but an incorrect answer, we manually corrected the answer. For Reasoning QA tasks, the corresponding reasoning steps were also corrected. (3) Pairs with unclear or confused questions were removed from the dataset. This final evaluation and correction process guarantees that every question in our benchmark is valid, and every answer is accurate. After removing the small number of invalid pairs, the final FinChart-Bench dataset consists of 7,016 high-quality QA pairs, distributed as follows: 2,383 True/False, 2,349 Multiple Choice, and 2,284 Reasoning QA samples. A detailed data analysis is provided in Appendix 2.3.

2.2 Human Evaluation

To ensure the highest standard of quality and accuracy, FinChart-Bench went through a two-stage human evaluation process, as depicted in Figure 1 (c and e). All manual annotations were conducted by the authors to maintain consistency and expertise throughout the project. The first stage of evaluation occurred in the Chart Filtering phase. This process only involved keeping or discarding the low quality chart. Given the straightforward nature of this task, the annotation rate was highly efficient, approximately 1,000 images per hour per annotator. The second stage was the QA Evaluation & Correction. This phase required a deep understanding of each chart and question, followed by careful verification and correction of the answers and reasoning steps. This complexity resulted in a slower pace, with an average of 80 QA pairs per hour per annotator. The entire human evaluation effort was substantial, spanning from early March to June 2025.

2.3 Data Analysis

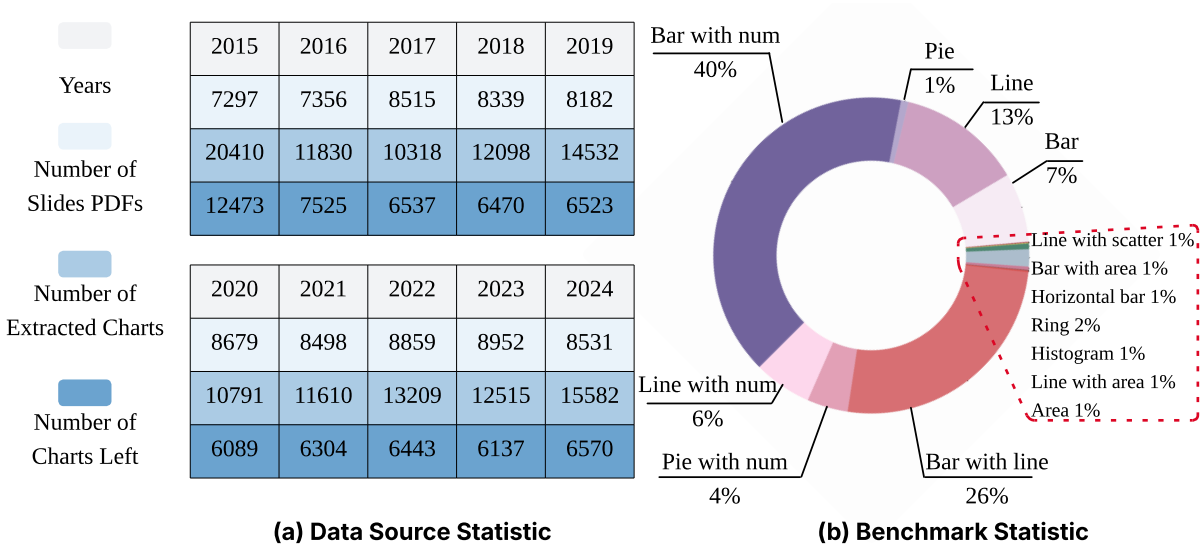
As Figure 2a illustrates, our data collection spans from 2015 to 2024. Starting with approximately 8,000 PDF slides per year, our Chart Extraction pipeline yielded over 10,000 charts annually, providing a large pool of candidates for downstream processing. Subsequent manual filtering reduced the count to around 7,000 charts per year, removing low-quality extractions and non-relevant visual elements. However, due to the extensive manual workload involved in QA Evaluation Correction, we keep only 120 charts from each year, resulting in a carefully curated subset for consistent annotation and analysis. As shown in figure 2b, our final benchmark comprises 14 chart types, with "Bar with num" charts dominating at 40%, reflecting its prevalence in real-world presentation materials. Examples of each benchmark chart type can be found in Appendix D.

3 Benchmarking Setting

3.1 Baseline

To establish a comprehensive performance benchmark on FinChart-Bench, we selected a diverse set of 26 prominent and state-of-the-art LVLMs for evaluation. These models are categorized into three groups: general-purpose open-source models, chart-specialized open-source models, and closed-source proprietary models.

General-purpose open-source LVLMs include 13 models: Llama-4 (Meta, 2025), Llama-3.2 (Meta, 2024), Llava-v1.6 (Liu et al., 2023b), Qwen2.5-VL (Team, 2025), Qwen2-VL (Wang et al., 2024a), Deepseek-vl2 (Wu et al., 2024b), Gemma-



(a) Data Source Statistic (b) Benchmark Statistic

Figure 2: Data statistics of FinChart-Bench. The tables on the left display the number of charts collected for each year from 2015 to 2024. The chart on the right illustrates the distribution of chart types in our benchmark, which includes a total of 14 distinct chart types.

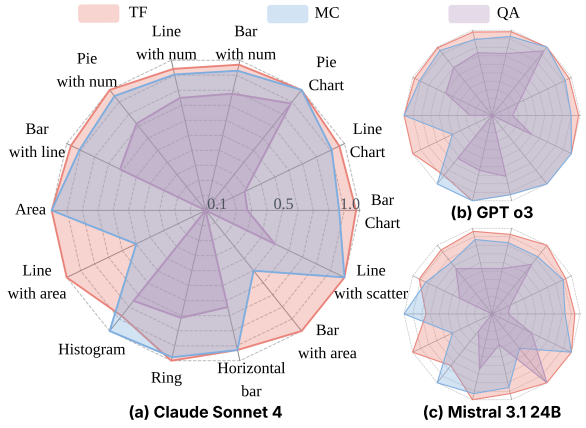


Figure 3: Detailed performance of three advanced models across different chart types. Additional results for other models are provided Figure 18 in the Appendix.

3 (Gemma, 2025), Mistral-3.1 (Mistral, 2025), Sa2VA-8B (Yuan et al., 2025), nanoVLM (Wiedmann et al., 2025), Cosmos-Reason1 (NVIDIA, 2025), MedGemma (Google, 2025b), Blip-2 (Li et al., 2023). We also include 5 chart-specialized open-source LVLMs: UniChart (Masry et al., 2023), Matcha (Liu et al., 2022), ChartGemma (Masry et al., 2024c), ChartInstruct-LLama2 (Masry et al., 2024a), ChartInstruct-FlanT5 (Masry et al., 2024b). Finally, we evaluated 7 closed-source models via respective APIs: GPT5, GPT4.1 (OpenAI, 2025a), GPT4.1-mini (OpenAI, 2025b), GPT4o (OpenAI, 2024), o3 (OpenAI, 2025d), o4-mini (OpenAI, 2025e), Claude Sonnet 4 (Anthropic, 2025), and Gemini 2.5 Pro (Google, 2025a). For each question in our benchmark, the input was combined with an instruction to let the model return its answer within a specific format: "Result = [[answer]]". This formatting ensures that the model's final answer can be reliably parsed from its output for automated evaluation. Details on the implementation are shown in Appendix 4.

3.2 Metric

A key motivation behind our benchmark is to eliminate the ambiguity commonly found in existing benchmarks. To this end, we design all ground truth answers to consist of a single token, making Exact Match (EM) an ideal evaluation metric due to its reliability and lack of ambiguity. In addition to EM, we introduce an Average (Avg.) score, which represents the weighted average of the model's scores across the three tasks, taking into account the number of questions in each.

$$\text{Avg.} = \frac{X \cdot \text{Score}_{\text{TF}} + Y \cdot \text{Score}_{\text{MC}} + Z \cdot \text{Score}_{\text{QA}}}{X + Y + Z}, \quad (1)$$

where Score_{TF}, Score_{MC}, and Score_{QA} denote the scores achieved by the model on the True/False, Multiple Choice, and Question Answering tasks, respectively, and X, Y, and Z indicate the number of questions in each task.

4 Implementation Details

To standardize the evaluation across all models, a specific prompting strategy was employed. For

Table 2: Model comparison on FinChart-Bench. The table is divided into three sections: comparisons between open-source models (left), chart-finetuned models (upper right), and closed-source models (lower right). We list the model sizes for all open-source models, as well as the input and output costs per 1M tokens for closed-source models. For each section and column, the highest score is shown in **Red**, and the second-highest score is **Blue**.

	TF	MC	QA	Avg.
Open Source Models				
LLaMa 3.2 (11B)	63.46	64.00	5.12	44.65
LLaVa v1.6 (7B)	63.30	36.94	1.84	34.47
Qwen2.5 VL (7B)	**95.09**	**82.81**	37.29	**72.16**
Qwen2 VL (7B)	91.86	19.40	24.77	45.76
DeepSeek VL2 (2.8B)	57.97	6.77	4.73	23.50
Gemma 3 (12B)	89.14	70.94	27.22	62.89
Mistral 3.1 (24B)	**92.95**	**84.17**	**44.90**	**74.37**
Sa2VA (8B)	86.62	75.79	19.43	61.12
nanoVLM (222M)	53.90	2.43	0.22	19.19
Cosmos (7B)	70.93	43.49	20.18	45.22
MedGemma (4B)	53.27	46.30	3.72	34.81
Blip 2 (2.7B)	49.79	0.77	0.0	17.17
LLaMa 4 (17B)	89.56	59.70	**59.78**	69.89

	TF	MC	QA	Avg.
Chart Fine-tuned Models				
UniChart (201M)	51.05	0.0	0.57	17.52
Matcha (282M)	49.54	0.0	0.70	17.05
ChartGemma (3B)	48.70	**0.29**	0.70	16.87
ChartInstruct-llama (7B)	**51.09**	0.17	**1.97**	**18.05**
ChartInstruct-T5 (3B)	**62.12**	**2.30**	**1.71**	**22.43**
Closed Source Models				
GPT 5 ($1.25, $10)	97.44	**92.13**	61.37	**83.92**
GPT 4.1 ($2, $8)	96.18	83.19	62.54	80.88
GPT 4o ($2.5, $10)	93.92	77.57	57.59	76.62
GPT 4.1 mini ($0.4, $1.8)	91.61	86.77	60.30	79.80
GPT o3 ($2, $8)	**97.86**	**92.17**	60.79	83.89
GPT o4 mini ($1, $4.4)	**97.61**	91.96	60.18	83.53
Claude Sonnet 4 ($3, $15)	96.94	91.66	**63.59**	**84.32**
Gemini 2.5 Pro ($1.25, $10)	97.27	89.70	**63.46**	83.73

each question in our benchmark, the input prompt was combined with an instruction to let the model return its answer within a specific format: "Result = [[answer]]". This formatting ensures that the model's final answer can be reliably parsed from its output for automated evaluation. All experiments, except for LLaMa 4, were conducted on a single NVIDIA A100 SXM4 GPU with 80GB of memory. LLaMa 4 was performed on 4 NVIDIA A100 SXM4 GPU with a total of 320GB memory. For all open-source models, we maintained their official repository configurations and loaded them with bfloat16 precision to optimize computational efficiency. Beyond this precision setting, no other modifications were made to the models' default parameters, ensuring a fair and reproducible comparison across all baseline evaluations.

5 Benchmarking Analysis

We have listed all 26 models results in Table 2 and have the following observations.

5.1 Average and Family Performance Gap

As shown in Table 2, most models perform well on the TF and MC tasks. Notably, leading open-source models like Mistral 3.1 (24B) and Qwen2.5 VL (7B) demonstrate performance that is competitive with, and in some cases exceeds, that of closed-source models like GPT-4o and GPT-4.1 mini. For instance, Mistral 3.1 achieves an MC score of 84.17%, surpassing GPT-4o's 77.57%. However, the QA task proves challenging for all models, with the top scores from both open-source (LLaMA 4 at 59.78%) and closed-source models (Claude Sonnet 4 at 63.59%) around 60%. This highlights a critical limitation in the complex reasoning abilities required for the QA task. We can also see in the Avg. column, open-source models such as Qwen2.5 VL (72.16%) and Mistral 3.1 (74.37%) achieve average performance scores that are close to those of closed-source models like GPT-4o (76.62%). While the average performance gap between the best open-source and closed-source models is narrowing, the primary challenge remains in advancing complex reasoning capabilities rather than excelling at simpler TF and MC tasks.

We also observe a *family performance gap*, where models from the same family exhibit substantial performance variance. This phenomenon is particularly evident among open-source models in the Qwen and LLaMa families. For instance, Qwen2.5 VL achieves 82.81% on the MC task, while Qwen2 VL scores only 19.40%. Similarly, LLaMa 4 achieves 59.78% on the QA task compared to just 5.12% for LLaMa 3.2. We view this family performance gap as a positive signal that these models are genuinely evolving across generations. In contrast, the closed-source GPT family (GPT-5, GPT-4.1, GPT-4o, GPT-4.1 mini, o3, and o4 mini) shows much more stable performance across tasks. This consistency may indicate that closed-source models are approaching a performance bottleneck on financial chart understanding tasks, with both older and newer versions achieving similar results. This potential bottleneck might also emerge in open-source models once they reach

today's leading closed-source models performance.

5.2 Performance Degradation in Upgraded Models

Although newer models within the same family generally achieve higher average performance than their predecessors, we observe instances of performance degradation on specific tasks. For example, LLaMa 4, as an upgraded version of LLaMa 3.2, demonstrates a higher overall average score (69.89% vs. 44.65%). However, on the MC task, LLaMa 3.2 outperforms LLaMa 4 (64.00% vs. 59.70%). A similar trend appears in the closed-source GPT family. While GPT-4.1 surpasses GPT-4.1 mini in average performance (80.88% vs. 79.80%), GPT-4.1 mini outperforms GPT-4.1 on the MC task (86.77% vs. 83.19%). This observation aligns with OpenAI's official report (OpenAI, 2025c), which notes that GPT-4.1 mini can sometimes outperform GPT-4.1 on the MathVista benchmark (Lu et al., 2023) and performs similarly on the CharXiv benchmark (Wang et al., 2024b), both involve chart visual reasoning. These findings support our earlier claim regarding a potential performance bottleneck. We believe that blindly upgrading a model or increasing its parameter size does not guarantee improved performance on all tasks. Future work should investigate the underlying causes of this bottleneck and explore strategies to overcome it.

5.3 Poor Instruction-Following Ability

Despite strong performance from some open-source models, many others, including those fine-tuned on chart data, struggle even with simple TF and MC tasks. Models such as DeepSeek-VL2, nanoVLM, and Blip2, as well as all models in the chart-fine-tuned category, failed to surpass 63% accuracy on the TF task and performed below 10% accuracy on the MC task. A closer analysis reveals that these poor results are largely due to weak instruction-following capabilities. When presented with complex multiple-choice instructions, especially those involving explicitly formatted outputs (e.g., Result = [[answer]]), these models often fail to generate properly structured answers or any response at all. Interestingly, most of these underperforming models have fewer than 3 billion parameters, suggesting that models below this size threshold may lack the capacity to handle chart question answering and reasoning tasks effectively. In contrast, models with over 7 billion parameters begin to show strong instruction-following behavior and significantly better task performance.

A key question arises: why do models specifically fine-tuned on chart data perform so poorly? We hypothesize that this may be due to overly aggressive fine-tuning, where the models are excessively optimized for narrow task performance, at the expense of their general instruction-following abilities inherited from the base LLM. This points to a critical trade-off between task-specific specialization and general language understanding, raising concerns about the design of fine-tuning pipelines for multimodal models.

5.4 Image Spatial Reasoning Ability

In addition to poor instruction-following capabilities, we also observe that many advanced models lack image spatial reasoning ability, the ability to accurately map visual chart components (e.g., bars, lines, or points) to their corresponding values or labels. As shown in Figure 3a, b, and c, we present the performance of leading models, both open-source and closed-source, on different chart types across all three tasks. While their performance varies by chart type, a consistent trend emerges in the QA task (highlighted in purple): all three models perform poorly on chart types such as "Line", "Bar", "Line with Scatter", "Area", and "Line with Area". In contrast, they achieve higher accuracy on "Line with Number" and "Bar with Number'.

This discrepancy stems from differences in how values are presented. In "Line with Number" and "Bar with Number" charts, numerical values are directly annotated next to the relevant visual components (e.g., next to a line point or on top of a bar), allowing the models to extract answers without performing spatial alignment. On the other hand, in standard "Line" and "Bar" charts, the corresponding values must be inferred from the axes. This requires models to spatially align chart elements (e.g., the top of a bar or a point on a line) with the appropriate value on the x- or y-axis. This is a task that demands a higher level of visual reasoning. This spatial alignment requirement also explains poor performance on other chart types such as "Area", "Line with Area", and "Line with Scatter". These results highlight that spatial reasoning remains a significant bottleneck in chart understanding tasks.

5.5 Limitations of Using LVLMs as Automatic Evaluators

A key reason we conducted two full rounds of manual evaluation for our benchmark is that current LVLMs are not reliable enough to serve as automated judges. This unreliability is demonstrated by both the chart filtering process (Figure 1c) and the detailed performance analysis of current LVLMs. After the chart extraction process (Third row in Figure 2a), one of the most advanced models, Qwen2.5 VL, initially extracted over 10,000 charts per year from financial documents. However, after manual filtering (fourth row), nearly half of the charts were discarded due to issues such as incompleteness, irrelevance, or low quality. These results highlight that despite strong performance in some areas, current LVLMs still struggle to consistently extract high-quality charts from real-world slide images.

Second, as discussed in earlier sections, our detailed performance analysis reveals that many models continue to struggle with fundamental abilities such as instruction following. Even the most capable LVLMs lack spatial reasoning ability. Furthermore, many advanced models exhibit similar weaknesses on specific chart types, suggesting a potential bottleneck in their chart understanding and reasoning capabilities. Together, these observations reinforce the need for human evaluation when building high-quality benchmarks.

6 More Benchmarking Results

6.1 Converging Performance Patterns in Advanced Models

As shown in Figure 3a, b, c, and Figure 18 in Appendix, an interesting trend emerges: lower-performing models, such as LLaVa v1.6, ChartInstruct, and Qwen2 VL, exhibit highly diverse performance patterns across tasks and chart types. In contrast, closed-source models and advanced open-source models like Mistral 3.1 and Qwen2.5 VL tend to follow similar performance patterns across chart types and tasks. For example, all of these advanced models show slightly lower accuracy on "Histogram" charts in the TF task, and consistently struggle with "Line with Area" charts in the MC task. In the QA task, they perform poorly on "Area", "Line with Area", "Line", and "Bar" charts. This convergence in performance patterns supports our earlier claim that advanced models lack strong image spatial reasoning ability. Moreover, it highlights a potential shared bottleneck that both open and closed-source models face when interpreting specific chart types. These findings suggest that future research should focus on improving model capabilities in spatially demanding visual reasoning tasks, particularly for the underperforming chart types identified.

6.2 Balancing Model Size and Cost for Real-World Applications

In Table 2, we report model sizes for all open-source and chart-fine-tuned models, and input/output costs per 1 million tokens for closed-source models. In the open-source category, although Mistral 3.1 slightly outperforms Qwen2.5 VL in average score (by just 2%), it does so with a much larger parameter size (24B vs. 7B). This suggests that Qwen2.5 VL achieves a better balance between size and performance, making it a more efficient choice for practical deployment.

For closed-source models, Claude Sonnet 4 achieves the highest average score (84.32%), followed closely by o3 (83.89%). However, their API prices are quite high, \$3/\$15 and \$2/\$8 (input/output per 1M tokens) respectively, making them less feasible for cost-sensitive real-world applications. In contrast, o4 mini, the second cheapest model, offers a compelling trade-off: it achieves an average score of 83.53%, only 1% lower than the top performer, while costing just \$1/\$4.4. Even the cheapest model GPT 4.1 mini, \$0.4/\$1.8, achieves a solid 79.80% average score, outperforming the more expensive GPT-4o (\$2.5/\$10). These findings suggest that o4 mini and GPT 4.1 mini strike the best balance between price and performance, making them strong candidates for real-world use. Given its higher cost and lower performance, GPT-4o may no longer be a practical choice for many applications.

7 Conclusions

In this paper, we present FinChart-Bench, a benchmark for evaluating LVLMs on real-world financial chart understanding. It overcomes issues in existing benchmarks, such as ambiguity, complexity, and weak validation. We designed single-token answers that still require deep reasoning, and underwent two rounds of manual review to ensure data quality. Evaluating 26 LVLMs revealed several insights, including degraded performance in newer models, poor instruction following in chart finetuned models, and limited spatial reasoning.

Limitations

Despite the high quality of FinChart-Bench, its scale remains limited due to the extensive manual filtering and correction processes required to ensure data accuracy and consistency. Processing all available data at once is challenging given the level of human validation involved. Nevertheless, we are committed to continuously contributing to the open-source community by releasing more high-quality labeled data in the future.

References

Anthropic. 2025. Claude 3.5 sonnet. https://www.anthropic.com/claude/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*.

Gemma. 2025. Gemma 3. https://huggingface.co/google/gemma-3-12b-it.

Akash Ghosh, Arkadeep Acharya, Sriparna Saha, Vinija Jain, and Aman Chadha. 2024. Exploring the frontier of vision-language models: A survey of current methodologies and future directions. *arXiv preprint arXiv:2404.07214*.

Google. 2025a. [link].

Google. 2025b. Medgemma hugging face. https://huggingface.co/google/medgemma-4b-it.

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*.

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

Anwen Hu, Yaya Shi, Haiyang Xu, Jiabo Ye, Qinghao Ye, Ming Yan, Chenliang Li, Qi Qian, Ji Zhang, and Fei Huang. 2024. mplug-paperowl: Scientific diagram analysis with the multimodal large language model. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6929–6938.

Kung-Hsiang Huang, Hou Pong Chan, Yi R Fung, Haoyi Qiu, Mingyang Zhou, Shafiq Joty, Shih-Fu Chang, and Heng Ji. 2024. From pixels to insights: A survey on automatic chart understanding in the era of large foundation models. *IEEE Transactions on Knowledge and Data Engineering*.

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, 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*.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. https://huggingface.co/Salesforce/blip2-opt-6.7b-coco.

Lei Li, Yuqi Wang, Runxin Xu, Peiyi Wang, Xiachong Feng, Lingpeng Kong, and Qi Liu. 2024. Multimodal arxiv: A dataset for improving scientific comprehension of large vision-language models. *arXiv preprint arXiv:2403.00231*.

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

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. https://huggingface.co/google/matcha-chartqa.

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

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023b. Improved baselines with visual instruction tuning. https://huggingface.co/llava-hf/llava-v1.6-mistral-7b-hf.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023c. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2023. Mathvista: Evaluating mathematical reasoning of

foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*.

Ahmed Masry, Parsa Kavehzadeh, Xuan Long Do, Enamul Hoque, and Shafiq Joty. 2023. Unichart: A universal vision-language pretrained model for chart comprehension and reasoning. *arXiv preprint arXiv:2305.14761*.

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

Ahmed Masry, Mehrad Shahmohammadi, Md Rizwan Parvez, Enamul Hoque, and Shafiq Joty. 2024a. Chartinstruct: Instruction tuning for chart comprehension and reasoning. https://huggingface.co/ahmed-masry/ChartInstruct-LLama2.

Ahmed Masry, Mehrad Shahmohammadi, Md Rizwan Parvez, Enamul Hoque, and Shafiq Joty. 2024b. Chartinstruct: Instruction tuning for chart comprehension and reasoning. https://huggingface.co/ahmed-masry/ChartInstruct-FlanT5-XL.

Ahmed Masry, Megh Thakkar, Aayush Bajaj, Aaryaman Kartha, Enamul Hoque, and Shafiq Joty. 2024c. Chartgemma: Visual instruction-tuning for chart reasoning in the wild. https://huggingface.co/ahmed-masry/chartgemma.

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

Meta. 2024. Llama-3.2-11b-vision-instruct. https://huggingface.co/meta-llama/Llama-3.2-11B-Vision-Instruct.

Meta. 2025. Llama-4-scout-17b-16e-instruct. https://huggingface.co/meta-llama/Llama-4-Scout-17B-16E-Instruct.

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.

Mistral. 2025. Mistral small 3.1. https://huggingface.co/mistralai/Mistral-Small-3.1-24B-Instruct-2503.

NVIDIA. 2025. Cosmos-reason1: From physical common sense to embodied reasoning. https://huggingface.co/nvidia/Cosmos-Reason1-7B.

OpenAI. 2024. Gpt-4o. https://platform.openai.com/docs/models/gpt-4o.

OpenAI. 2025a. Introducing gpt-4.1 in the api. https://platform.openai.com/docs/models/gpt-4.1.

OpenAI. 2025b. Introducing gpt-4.1 in the api. https://platform.openai.com/docs/models/gpt-4.1-mini.

OpenAI. 2025c. Introducing GPT-4.1 in the API — openai.com. https://openai.com/index/gpt-4-1/. [Accessed 09-07-2025].

OpenAI. 2025d. Introducing openai o3 and o4-mini. https://platform.openai.com/docs/models/o3.

OpenAI. 2025e. Introducing openai o3 and o4-mini. https://platform.openai.com/docs/models/o4-mini.

Dong Shu, Haiyan Zhao, Jingyu Hu, Weiru Liu, Ali Payani, Lu Cheng, and Mengnan Du. 2025. Large vision-language model alignment and misalignment: A survey through the lens of explainability. *arXiv preprint arXiv:2501.01346*.

Qwen Team. 2025. Qwen2.5-vl.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024a. 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. 2024b. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. *Advances in Neural Information Processing Systems*, 37:113569–113697.

Luis Wiedmann, Aritra Roy Gosthipaty, and Andrés Marafioti. 2025. nanovlm. https://github.com/huggingface/nanoVLM.

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. *arXiv preprint arXiv:2405.07001*.

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. 2024b. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. https://huggingface.co/deepseek-ai/deepseek-vl2-small.

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*.

Peng Xu, Wenqi Shao, Kaipeng Zhang, Peng Gao, Shuo Liu, Meng Lei, Fanqing Meng, Siyuan Huang, Yu Qiao, and Ping Luo. 2024. Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

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*.

Haobo Yuan, Xiangtai Li, Tao Zhang, Zilong Huang, Shilin Xu, Shunping Ji, Yunhai Tong, Lu Qi, Jiashi Feng, and Ming-Hsuan Yang. 2025. Sa2va: Marrying sam2 with llava for dense grounded understanding of images and videos. https://huggingface.co/ByteDance/Sa2VA-8B.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. 2024. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Bingchen Zhao, Yongshuo Zong, Letian Zhang, and Timothy Hospedales. 2024. Benchmarking multi-image understanding in vision and language models: Perception, knowledge, reasoning, and multi-hop reasoning. *arXiv preprint arXiv:2406.12742*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Related Work

A.1 Large Vision Language Model

Large vision-language models are a subclass of multimodal large language models (MLLMs) that jointly process visual and textual information, typically taking images and text as input and generating text as output (Zhu et al., 2023; Liu et al., 2023c). Early LVLMs focused on general purpose tasks such as image captioning and visual question answering (VQA), which served as foundational benchmarks to assess whether these models could "see" and describe images in natural language (Zhang et al., 2024; Ghosh et al., 2024). As model capacity increased and alignment techniques improved, researchers began probing deeper capabilities of LVLMs to assess whether they could truly "understand" visual inputs (Shu et al., 2025). This shift led to the development of more specialized tasks and benchmarks beyond surface-level description, such as visual commonsense reasoning and scientific figure understanding (Zhao et al., 2024). Among these directions, chart understanding emerged as a particularly challenging and informative task. Unlike natural images, charts encode structured information such as axes, labels, legends, and quantitative relationships. Understanding a chart often requires not just visual recognition, but also domain-specific reasoning in fields such as science, mathematics, or finance. As such, chart understanding is a valuable direction deserving increased attention from the research community.

A.2 Difference with Other Chart Benchmark

A number of benchmarks have been developed to evaluate the chart understanding capabilities of various models. Early efforts, such as FigureQA (Kahou et al., 2017), DVQA (Kafle et al., 2018), and PlotQA (Methani et al., 2020), primarily relied on template-based questions and synthetic data, which limited the complexity and real-world applicability of the tasks. More recent benchmarks, including Chart-to-text (Kantharaj et al., 2022), ChartBench (Xu et al., 2023), and ChartX (Xia et al., 2024), have made significant strides by incorporating more diverse chart types and free-form questions generated from real-world data. However, this move towards complexity has introduced new challenges. The ground-truth answers in many of these benchmarks are often long and complex, making them difficult to evaluate with standard metrics like Exact Match or BLEU. This has led to a reliance on costly, and sometimes inconsistent, GPT-based evaluation, deviating from the goal of creating easily reproducible and accessible benchmarks. Furthermore, a critical review reveals that most existing datasets only have a small subset of their data checked by human, which fails to guarantee the overall quality and accuracy of the benchmark.

In contrast, our FinChart-Bench is designed to address these specific shortcomings. First, to ensure data quality, our entire benchmark underwent a rigorous two-stage manual evaluation process. Every chart image was first inspected for visual quality and completeness. Subsequently, each question-answer pair was manually reviewed and corrected to eliminate errors and ambiguity. Second, while our free-form questions are designed to demand sophisticated reasoning skills, the corresponding ground-truth answers are still concise and unambiguous. This unique combination allows for evaluation using the straightforward and reliable Exact Match metric, removing the need for expensive API-based assessments. Finally, FinChart-Bench is distinguished by its specific focus on the finance domain, providing a specialized and challenging new dataset for testing model performance on complex, real-world financial charts.

B Tasks Design Choice

The design of the task types in FinChart-Bench were deliberately guided by a core principle, which is balancing the task difficulty with evaluation simplicity. Our primary objective was to ensure that the ground-truth answer for every task could be distilled into a single, unambiguous token, thereby minimizing subjectivity and complexity during evaluation, and supporting reliable, automated scoring at scale. To this end, we incorporated two foundational "closed-ended" tasks: True/False and Multiple Choice. These formats are naturally suited to our design principle, as their answers are inherently single-token, and their discrete option spaces reduce ambiguity in both annotation and evaluation. However, to assess the model's reasoning abilities more deeply, we also included an "open-ended" task: Reasoning QA. The Reasoning QA task is specifically designed to test a model's ability to interpret chart data and perform numerical calculations, including multi-step comparisons, aggregations, and value derivations implied by the chart. To maintain evaluation simplicity, we ensure its answers are also single-token, enabling con-

sistent scoring without post-processing heuristics. We format the answer as a floating-point number without any punctuation or units, enforcing a strict and machine-checkable output constraint. The required unit is always specified within the question itself, so the model can focus on computation rather than formatting decisions. The reasoning process is stored separately as metadata and is not part of the ground-truth answer used for evaluation, allowing us to preserve explanatory supervision without introducing evaluation subjectivity. This design choice allows our benchmark to rigorously test complex reasoning while enabling straightforward, objective evaluation, and making results comparable across models and settings.

C Examples of the Prompt Used

C.1 Prompt Used in Chart Extraction

As illustrated in Figure 1b, we use the "Qwen2.5-VL-7B-Instruct" model to extract all charts from financial slide PDFs. The extraction is guided by the following prompt:

Chart Position Detection

Task: Outline the full position of each complete chart in the image, ensuring the position encompasses all essential elements: title, axes, labels, legends, data visualizations, etc.

Output Format: All coordinates in JSON format as a list of dictionaries. Follow this exact format:

`[{'bbox_2d': [x_min, y_min, x_max, y_max], 'label': 'chart'}, ...]`

Note: If no charts are detected, simply return `'No Chart'`.

C.2 Prompt Used in QA Generation

As shown in Figure 1d, we use GPT-4.1 to generate three types of question–answer pairs based on each financial chart. Below, we present the prompt used during this process.

True/False Question Generation

You are a helpful assistant that answers in JSON format with two True/False questions: one 'True', one 'False'.

You are given an image. Generate **two** True or False questions based on the image, one with a **True** answer and one with a **False** answer.

Response Format: Respond strictly in the following JSON format:

```
{
  "question1": {
    "question": "<true_question_text>"
      ,
    "answer": "True"
  },
  "question2": {
    "question": "<false_question_text
      >",
    "answer": "False"
  }
}
```

Multiple-Choice Question Generation

You are a helpful assistant that analyzes finance chart images and generates multiple-choice questions strictly in a structured JSON format.

You are given a finance-related chart image. Based on the visual data and financial concepts presented in the chart, generate **two** multiple-choice questions. Each question must have four answer choices (A, B, C, D) with **only one correct answer**.

Requirements: The questions should:

- Be clearly related to the financial insights or data trends visible in the chart.
- Not require external information outside what is shown in the chart.

Response Format: Return your response strictly in the following JSON format:

```
{
  "question1": {
    "question": "<question_text_1>",
    "choices": {
      "A": "<choice_text>",
      "B": "<choice_text>",
      "C": "<choice_text>",
      "D": "<choice_text>"
    },
    "answer": "<correct_option_letter
       >"
  },
  "question2": {
    "question": "<question_text_2>",
    "choices": {
      "A": "<choice_text>",
      "B": "<choice_text>",
      "C": "<choice_text>",
      "D": "<choice_text>"
    },
    "answer": "<correct_option_letter
       >"
  }
}
```

Question Answering Question Generation

You are a helpful assistant that analyzes finance-related chart images and generates structured, reasoning-based quantitative questions in JSON format.

You are given a finance-related chart image. Based on the data and trends presented in the chart, generate **two** quantitative question-answer pairs. Each question must require **numerical reasoning or calculation**, and the answer must be a **number** (not text or a choice). For each question, also provide a clear explanation of the reasoning or calculation used to arrive at the answer.

Guidelines:

- Base your questions solely on the information presented in the chart.
- Ensure each question involves basic computation (e.g., growth rates, differences, percentages, trends).
- Do not use any external information not shown in the chart.

Response Format: Return your response strictly in the following JSON format:

```
{
  "question1": {
    "question": "<question_text_1>",
    "reasoning": "<reasoning_process_1
       >",
    "answer": <numeric_answer_1>
  },
  "question2": {
    "question": "<question_text_2>",
    "reasoning": "<reasoning_process_2
       >",
    "answer": <numeric_answer_2>
  }
}
```

D Examples of Different Chart Types

Figures 5 to 17 present example charts for all 14 chart types included in our benchmark.

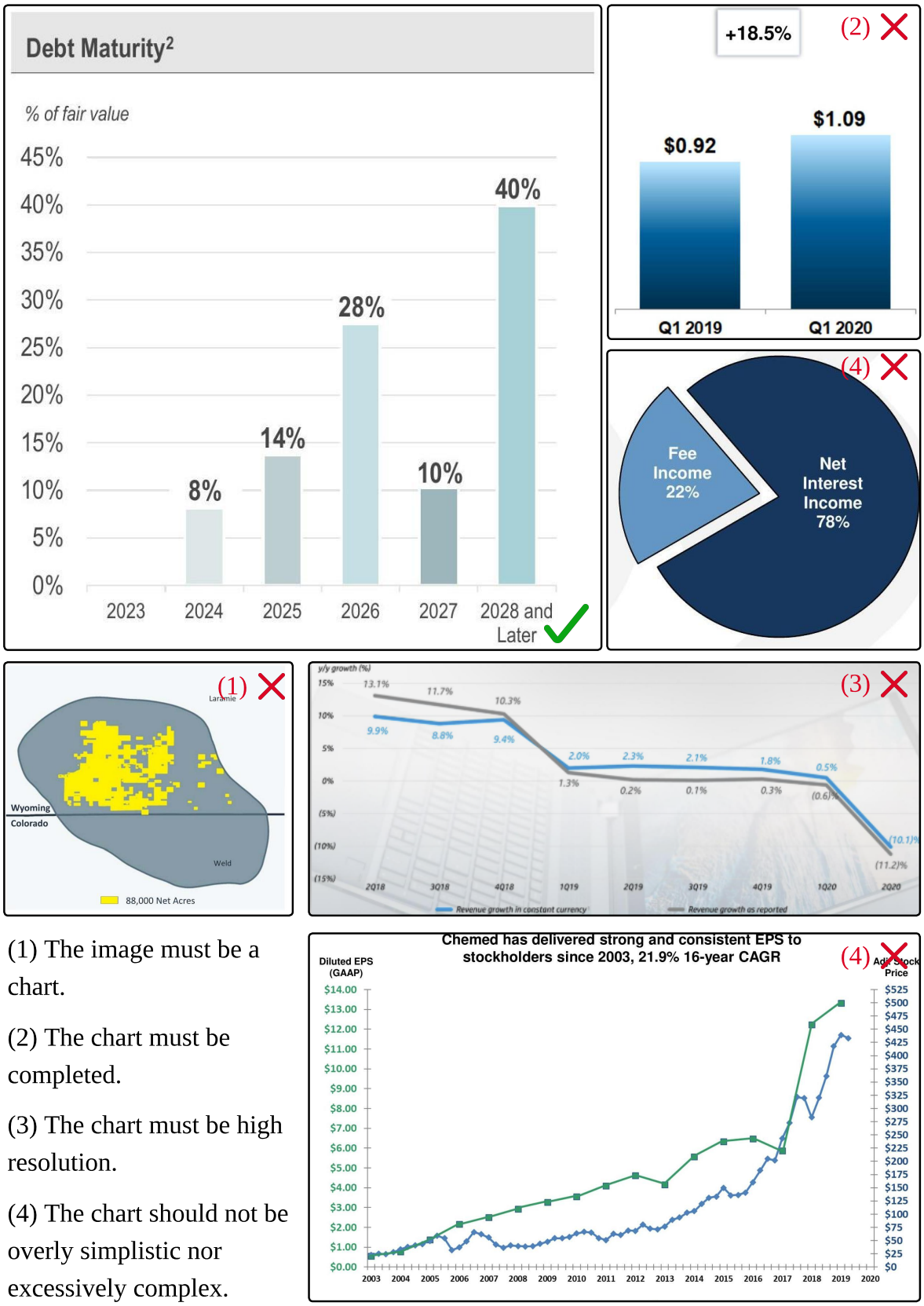


Figure 4: Examples of the chart filtering process. The four filtering criteria are listed in the lower left, and each image is annotated with the specific criterion it fails to meet.

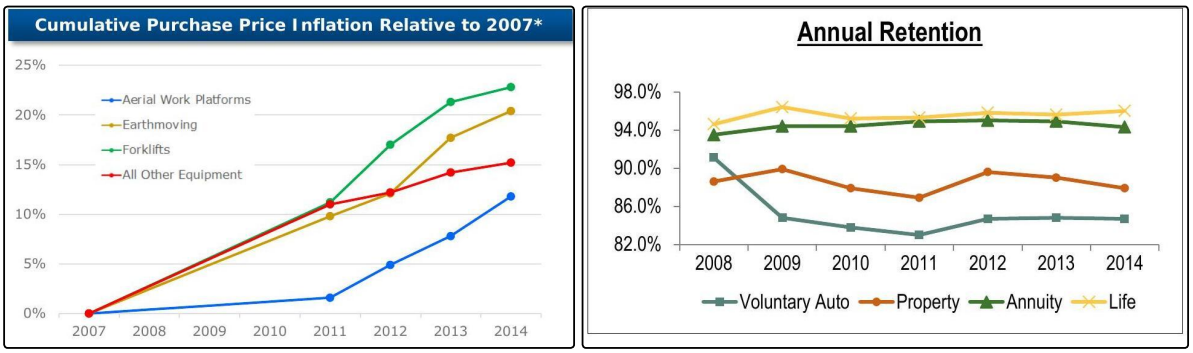


Figure 5: Examples of the chart type “Line”.

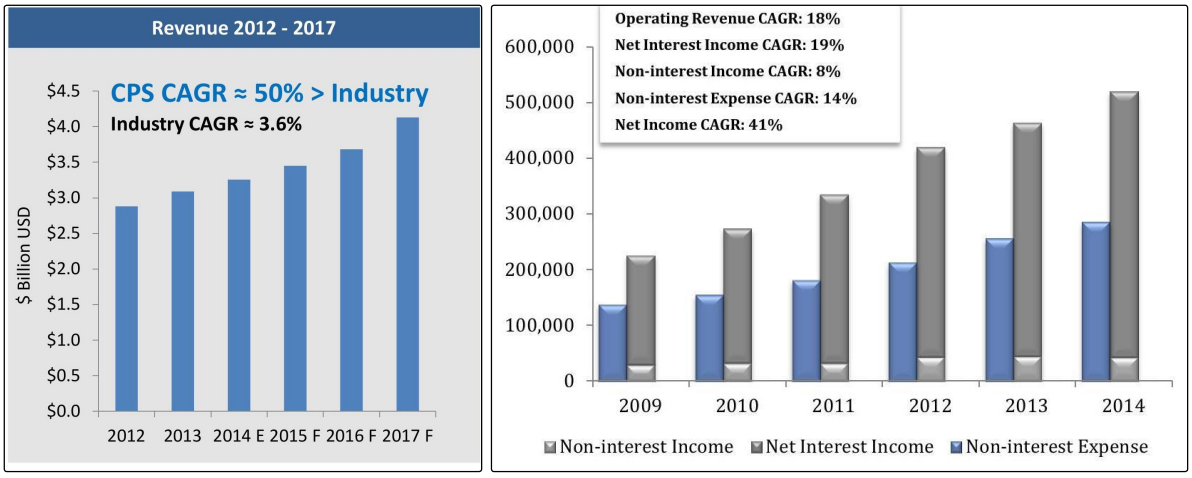


Figure 6: Examples of the chart type “Bar”.

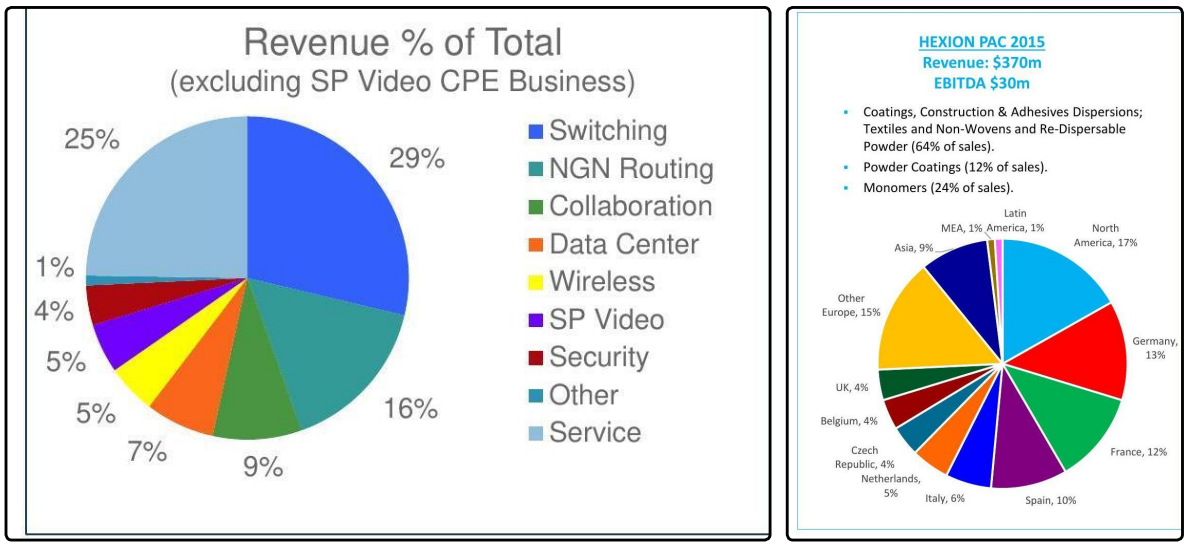


Figure 7: Examples of the chart type “Pie”.

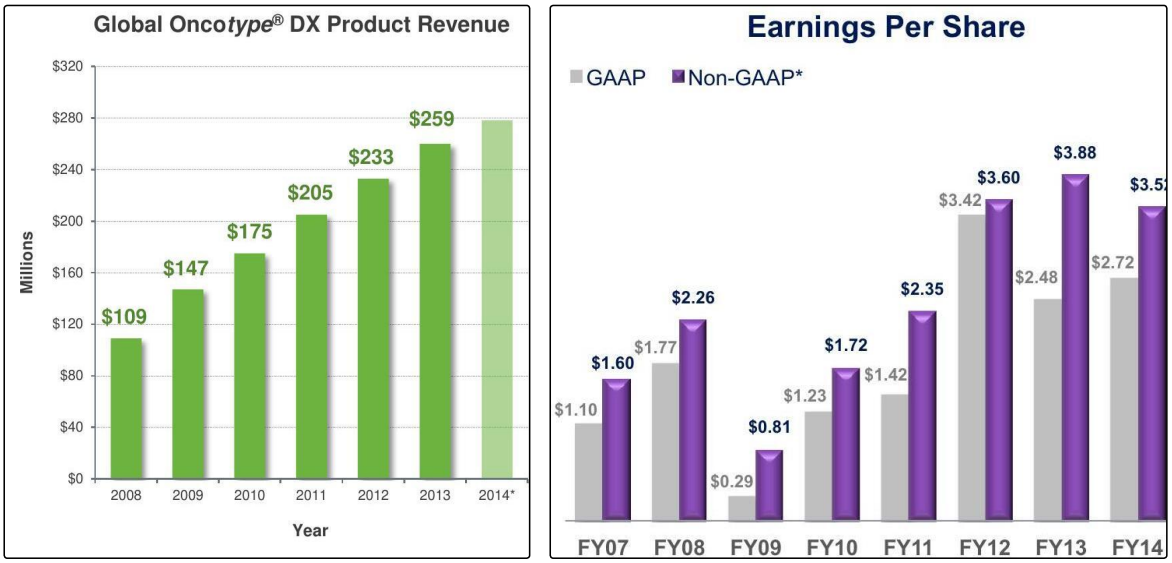


Figure 8: Examples of the chart type “Bar with num”.

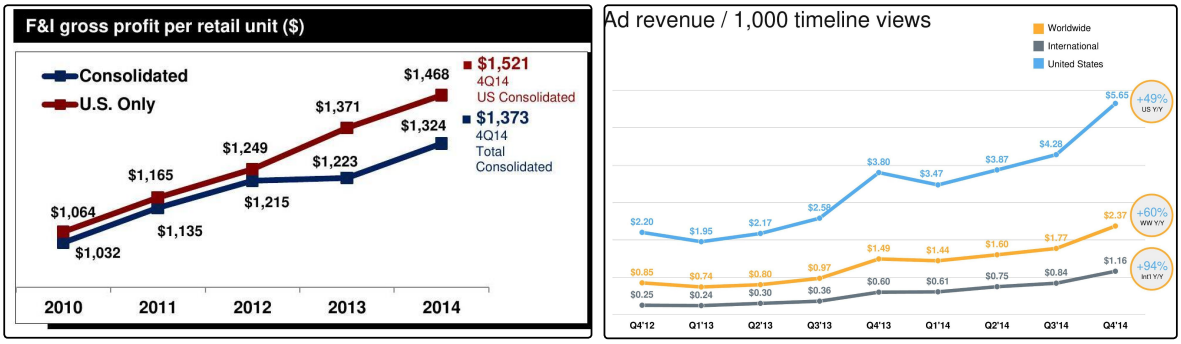


Figure 9: Examples of the chart type “Line with num”.

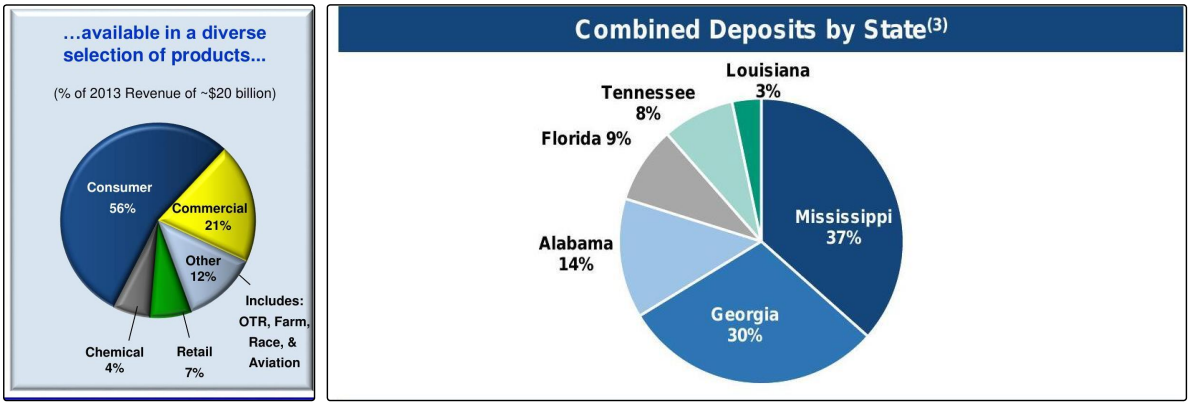


Figure 10: Examples of the chart type “Pie with num”.

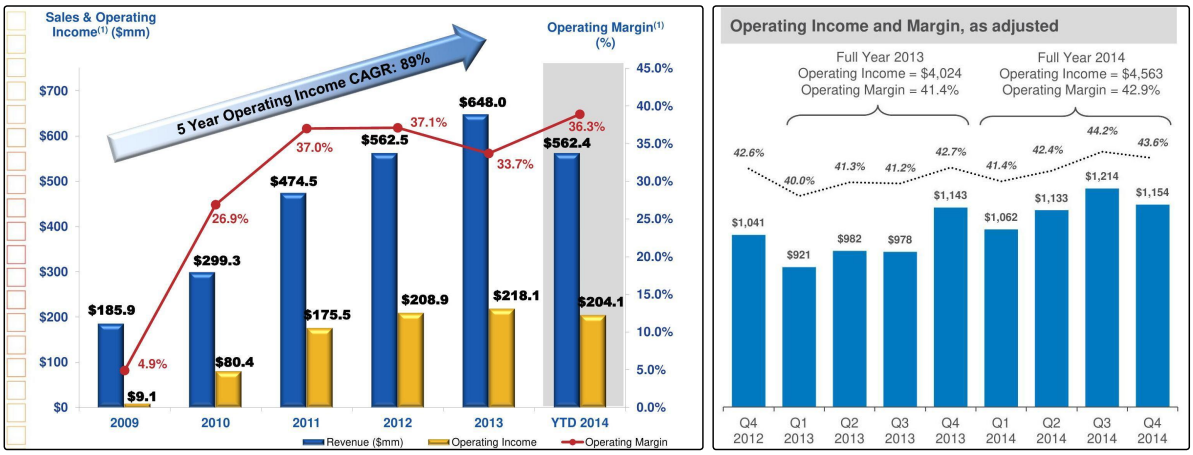


Figure 11: Examples of the chart type “Bar with line”.

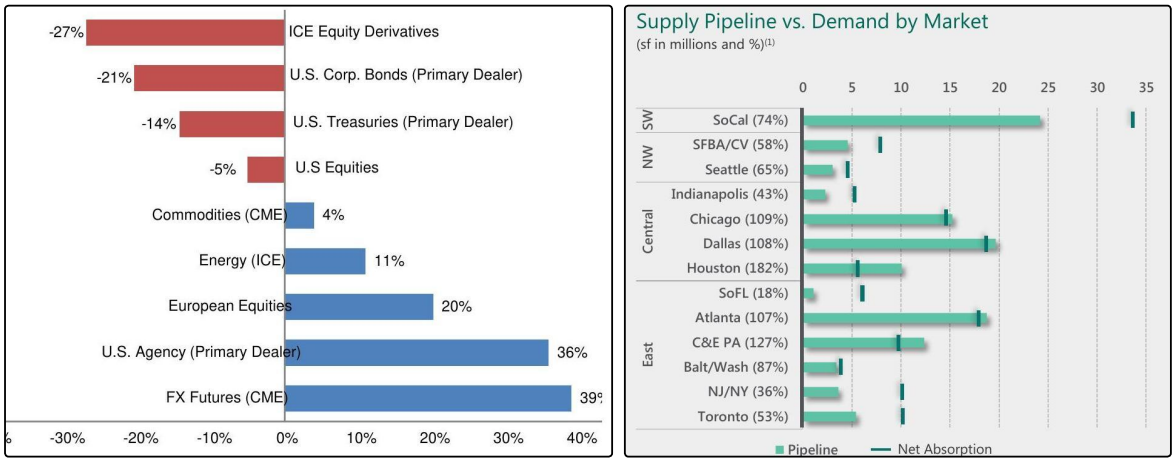


Figure 12: Examples of the chart type “Horizontal bar”.

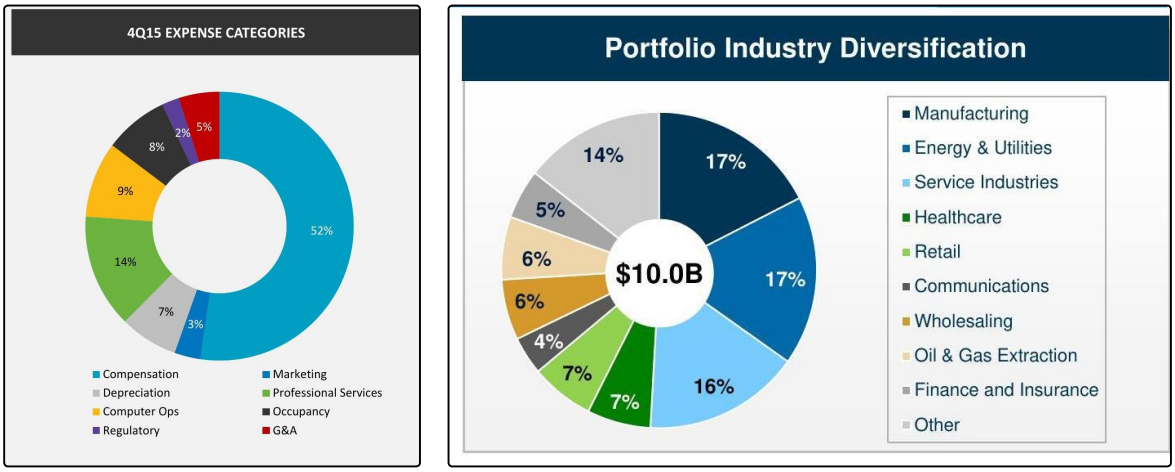


Figure 13: Examples of the chart type “Ring”.

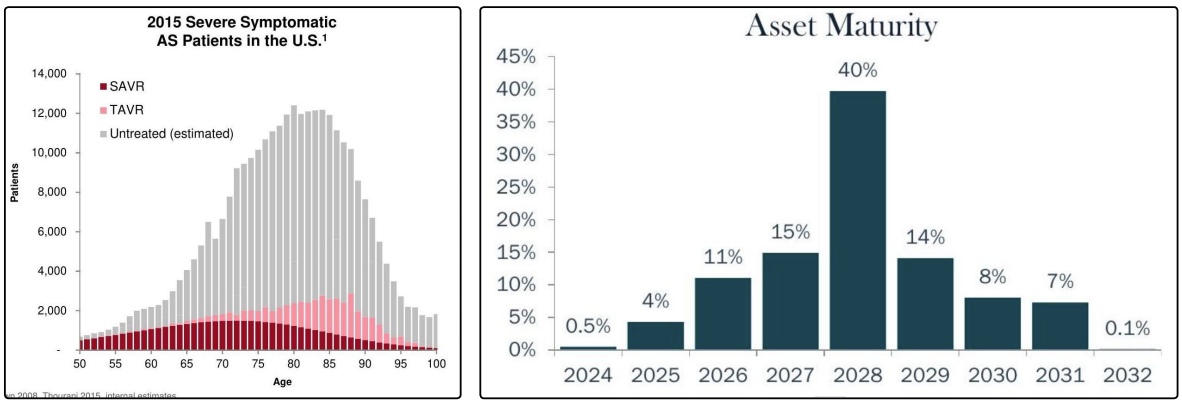


Figure 14: Examples of the chart type “Histogram”.

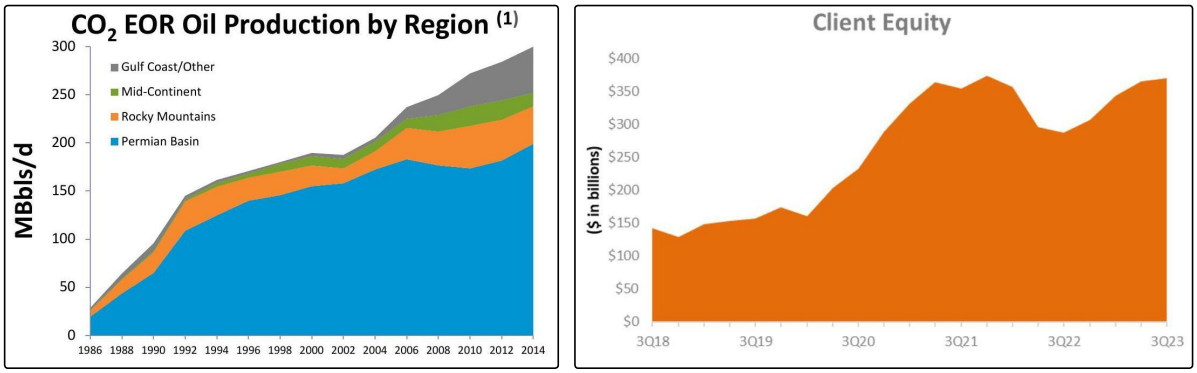


Figure 15: Examples of the chart type “Area”.

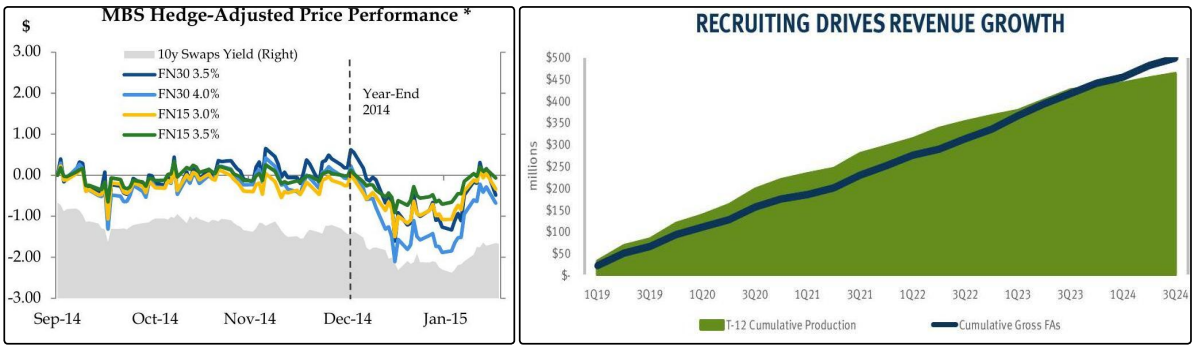
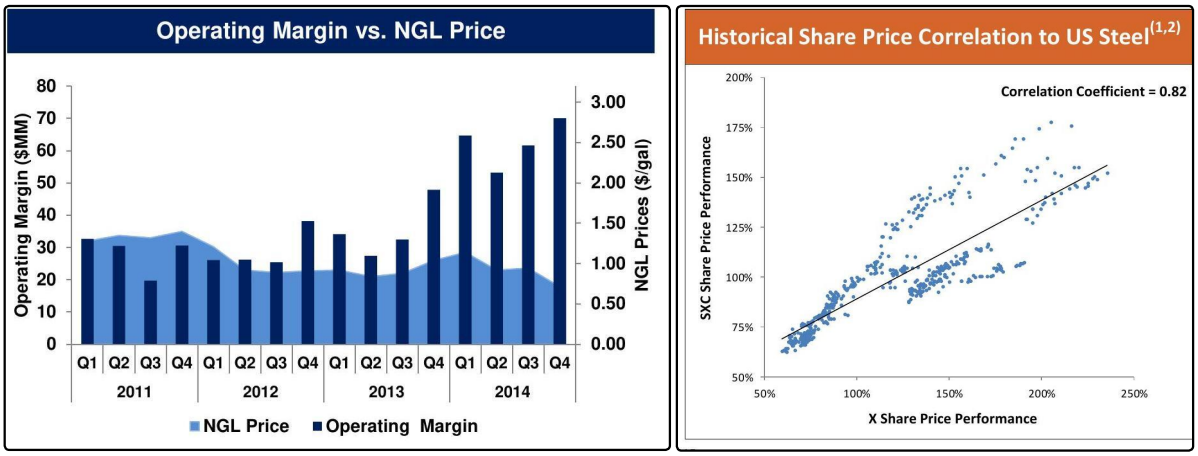


Figure 16: Examples of the chart type “Line with area”.



(a) Bar with area **(b) Line with scatter**

Figure 17: Examples of the chart type “Bar with area” (Left), and “Line with scatter” (Right).

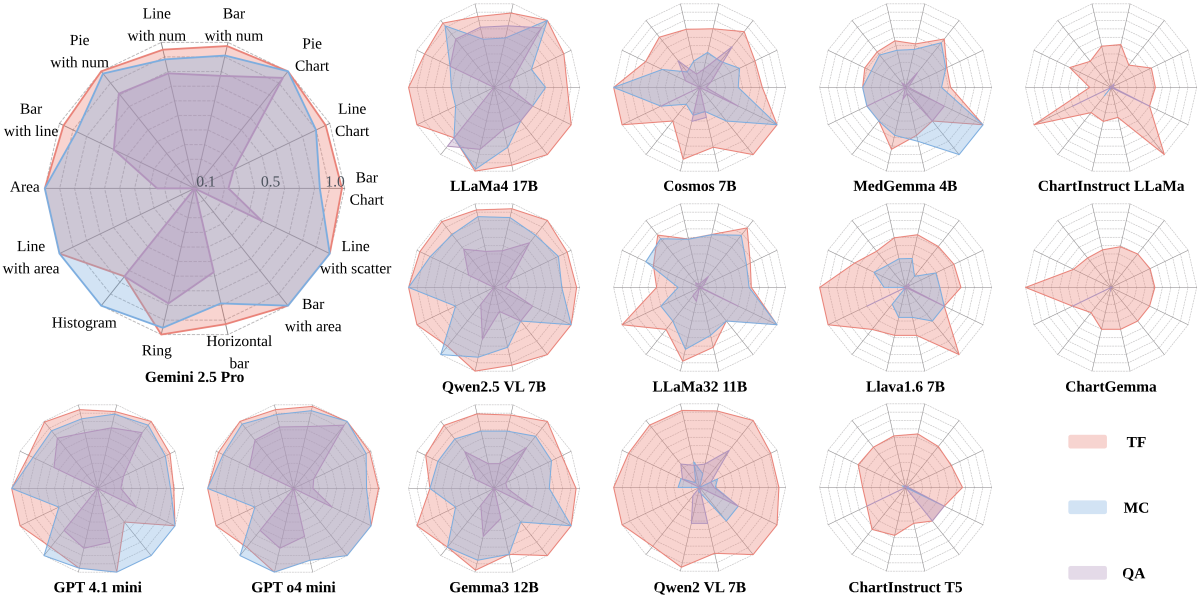


Figure 18: Detailed performance of all models across different chart types