ChartDesign: TOWARDS LLM DESIGNER OF DATA VISUALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Charts require many design choices, and the same data table can support multiple reasonable visualizations depending on the intended task and audience. We introduce ChartDesign, a framework that post-trains compact language models to predict a renderer-neutral chart-design specification from tabular data. We construct 2,118 human-authored chart examples from PewResearch and CharXiV, reserve 200 manually verified and corrected charts for evaluation, and audit 300 training examples to characterize annotation quality. Post-training substantially improves reference-match accuracy over matched zero-shot, rule-based, and direct-copy retrieval baselines, with Phi-3 full fine-tuning reaching 82.0% on the combined human-verified evaluation set. Human evaluation further finds high visual plausibility and frequent preference for generated charts, while showing that reference agreement and design quality are distinct. These results support learning reusable human-authored chart-design preferences without tying the output representation to a single rendering library.

1 INTRODUCTION

Data visualizations are central to scientific communication, journalism, and exploratory analysis. Producing an effective chart, however, requires more than selecting a plotting library. A designer must choose an appropriate chart family, decide how values are grouped or stacked, select orientation and visual encodings, place legends and labels, tune spacing and grid lines, and balance readability against visual complexity. Existing plotting tools expose these controls, but much of this design process remains manual.

Recent language and vision-language models can parse charts, extract values, generate plotting code, and recommend visualizations. These capabilities motivate a complementary question: *can a language model learn recurring chart-design preferences from human-authored examples and predict a structured design specification directly from a data table?* A structured intermediate representation is useful because it separates design decisions from renderer-specific program syntax. The same specification can, in principle, be interpreted by different visualization libraries while retaining the intended chart family, orientation, labels, legend behavior, and mark-level layout.

The task is inherently underdetermined. A single table may support several defensible visualizations depending on the analyst's objective, intended audience, and communication context. We therefore do not treat the source chart as the uniquely correct design. Instead, our quantitative metric measures agreement with one human-authored reference, while a separate human evaluation assesses readability, design appropriateness, visual clarity, information preservation, and pairwise preference. This distinction is important: stylistic disagreement with a reference does not necessarily imply a poor visualization.

We introduce ChartDesign, which maps a CSV table to a hierarchical JSON specification covering chart family and subtype, orientation, text elements, axes, legends, grid lines, and graphical-mark layout. We construct 2,118 chart examples from two complementary sources: PewResearch, which contains presentation-oriented public visualizations, and CharXiV, which contains scientific figures. A vision-language model extracts values visibly encoded in each chart and generates an initial design JSON. We hold out 100 charts from each source and manually inspect and correct their extracted data and design annotations, producing a 200-chart human-verified evaluation set. We also audit 300 training examples to quantify the quality of automatically generated supervision.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Our contributions are:

- We formulate chart-design prediction as structured, renderer-neutral generation from tabular data, explicitly separating reference agreement from universal visualization correctness.
- We construct a 2,118-chart corpus spanning presentation-oriented and scientific visualizations, together with a 200-chart human-verified evaluation set and a 300-chart annotation-quality audit.
- We evaluate compact open-weight models under matched zero-shot and post-training settings, with rule-based and retrieval controls, attribute-level analysis, and training-data scaling.
- We complement reference-based metrics with a blinded human study that evaluates both fidelity to a source design and visualization quality using criteria that do not require exact reference matching.

2 Related Work

Chart understanding and extraction. Chart understanding includes classification, structure parsing, question answering, and recovery of underlying values. ChartOCR Luo et al. (2021) uses a hybrid vision pipeline for extracting structured values from chart images, while multimodal models such as LLaVA Liu et al. (2023) support broader visual-language reasoning. Datasets such as ChartQA Masry et al. (2022), FigureQA Kahou et al. (2018), and DVQA Kafle et al. (2018) focus primarily on reasoning or question answering over charts. These directions are complementary to our setting: at inference time, `ChartDesign` receives a data table and predicts design choices rather than recovering data from an image.

Visualization recommendation. Classical systems such as APT Mackinlay (1986), SAGE Roth et al. (1994), ShowMe Mackinlay et al. (2007), and Voyager Wongsuphasawat et al. (2016) encode or operationalize visualization-design heuristics. Learning-based systems infer recommendations from data–visualization examples. Data2Vis Dibia & Demiralp (2019) generates Vega-Lite specifications, and AdaVis Zhang et al. (2024) explicitly models the one-to-many relationship between a dataset and multiple appropriate visualizations. That one-to-many framing is closely related to our treatment of chart design as underdetermined rather than as a single-label prediction problem.

LLMs for chart generation and feedback. ChartGPT Tian et al. (2024), Chat2Vis Maddigan & Susnjak (2023), and related systems use language models to generate plotting code. C^2 Koh et al. (2025) studies reference-free auto-feedback for iterative chart generation. Our focus differs in the output representation: instead of generating renderer-specific code, `ChartDesign` predicts an intermediate design JSON that can be interpreted by multiple plotting libraries.

Chart datasets and structured representations. VisText Tang et al. (2023) provides rasterized charts, backing data tables, scene graphs, and Vega-Lite specifications for chart captioning. Its structured information overlaps with several fields in our representation, so our distinction is not the mere availability of structured chart data. Rather, we study design-specification prediction from tabular input and evaluate agreement with human-authored designs together with perceptual quality. CharXiV Wang et al. (2024) evaluates realistic chart understanding and supplies one of the source domains in our corpus. Appendix A.4 summarizes the available information and primary objectives of representative systems.

3 ChartDesign

3.1 Problem formulation and design representation

Given a tabular input x, our goal is to predict a structured design specification y describing how the table can be visualized. The specification is represented as hierarchical JSON and includes chart family and subtype, orientation, text elements, axis configuration, legend properties, grid lines, and mark-level layout. The representation is library-neutral: fields describe intended visual properties rather than Matplotlib-, Vega-Lite-, Altair-, or ggplot2-specific syntax. Renderer neutrality therefore means portability of the specification, not pixel-identical rendering across libraries.

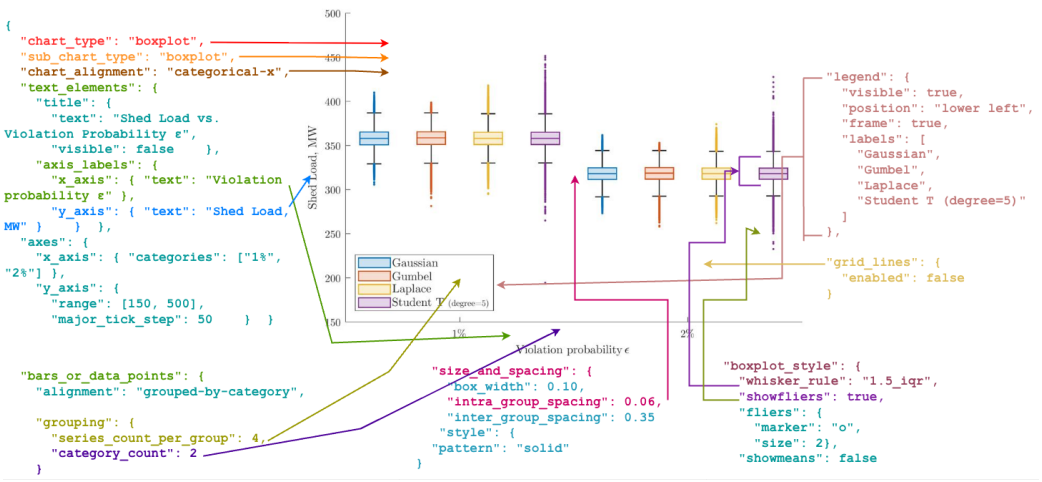

Figure 1: **Design schema overview.** A sample box plot is annotated with selected fields from the structured design JSON. Arrows connect chart components to schema attributes such as chart family, text elements, axes, legend, grid lines, mark layout, spacing, and box-plot style.

Figure 1 illustrates how fields in the JSON correspond to components of a human-authored chart. Attributes that do not apply to a chart family are omitted. For example, pie charts do not require Cartesian axes, while box plots expose family-specific parameters such as whisker rules and flier visibility. The complete machine-readable schema is included with the submission and summarized in Appendix A.7.

Figure 2 illustrates how the same structured specification can be interpreted by multiple rendering libraries. The purpose of this comparison is portability rather than pixel-level equivalence: high-level intent such as chart family, axes, legend semantics, and series structure can be transferred, while library defaults can still change marker appearance, spacing, and exact legend placement.

3.2 Corpus Construction

Our corpus contains 2,118 human-authored charts: 1,101 from PewResearch (Pew Research Center, 2015; Masry et al., 2022) and 1,017 from CharXiV (Wang et al., 2024). The two sources provide complementary design distributions, with presentation-oriented public visualizations in PewResearch and scientific plots in CharXiV. We reserve 100 charts from each source for human-verified evaluation, leaving 1,918 examples for the training and validation pool. Appendix A.5 summarizes the verified source totals and schema-domain diversity.

For each source chart, `Phi-3.5-Vision Instruct` extracts the values visibly encoded in the figure as one or more CSV tables. The extraction target is limited to plotted values or displayed summary statistics; latent raw observations that cannot be recovered from the published chart are outside scope. The source image and extracted table are then used to generate the initial design JSON. Figure 3 summarizes this process. The 200 held-out evaluation examples are manually verified and corrected for both CSV content and design JSON before scoring.

3.3 Training-Annotation Audit

Because most training labels are generated automatically, we audit 300 training examples independently of the held-out test set. For recoverable numerical cells that can be aligned with the human-corrected CSV, normalization to the corresponding reference range gives NMAE 0.034 and NRMSE 0.057. These metrics are conditional on aligned numeric cells and do not measure structural failures such as unmatched rows, missing series, or an incorrectly extracted table.

Across design attributes, the automatically generated JSONs achieve 92.1% average agreement with the audit labels, and inter-annotator reliability for the human audit labels is Krippendorff's $\alpha = 0.84$.

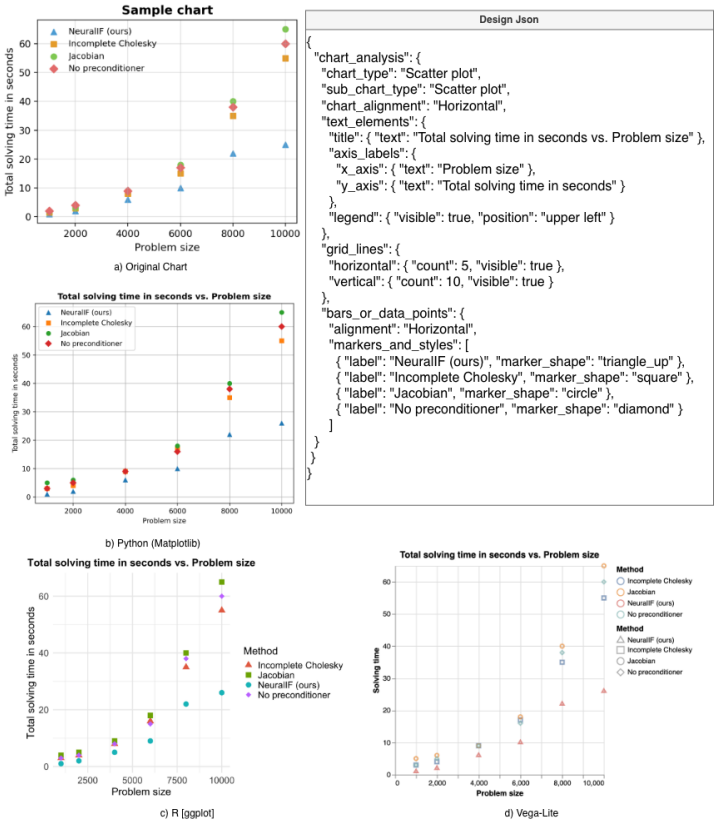

Figure 2: **Cross-renderer portability of one structured design specification.** The same design JSON is interpreted by Matplotlib, Vega-Lite, Altair, and ggplot2. High-level design intent is transferable, while low-level rendering details can differ across libraries.

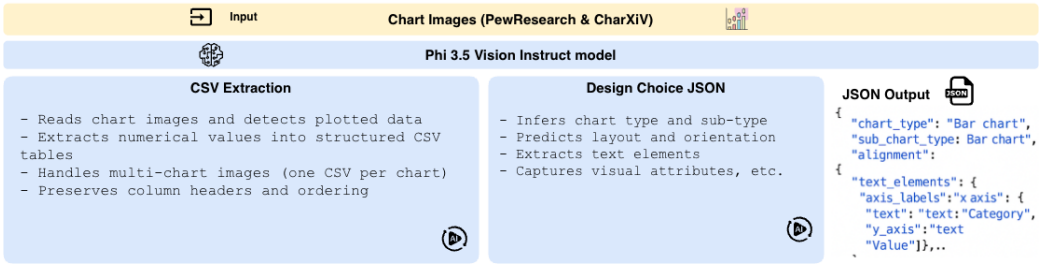

Figure 3: **Dataset construction and annotation pipeline.** Human-authored chart images are processed by a vision-language model to recover visible data as CSV and infer a structured design JSON.

Structural attributes such as chart family and orientation show higher agreement than fine-grained spacing or grid-line properties. Appendix A.6 provides the attribute-level audit and clarifies the scope of each metric.

3.4 POST-TRAINING

The primary reported backbones are Phi-3-mini-4k-instruct (4.2B) and Qwen-3 (8B). Each example contains an instruction and CSV table as input and the corresponding design JSON as the target. We optimize standard autoregressive cross-entropy. For LoRA, the pretrained weights are frozen

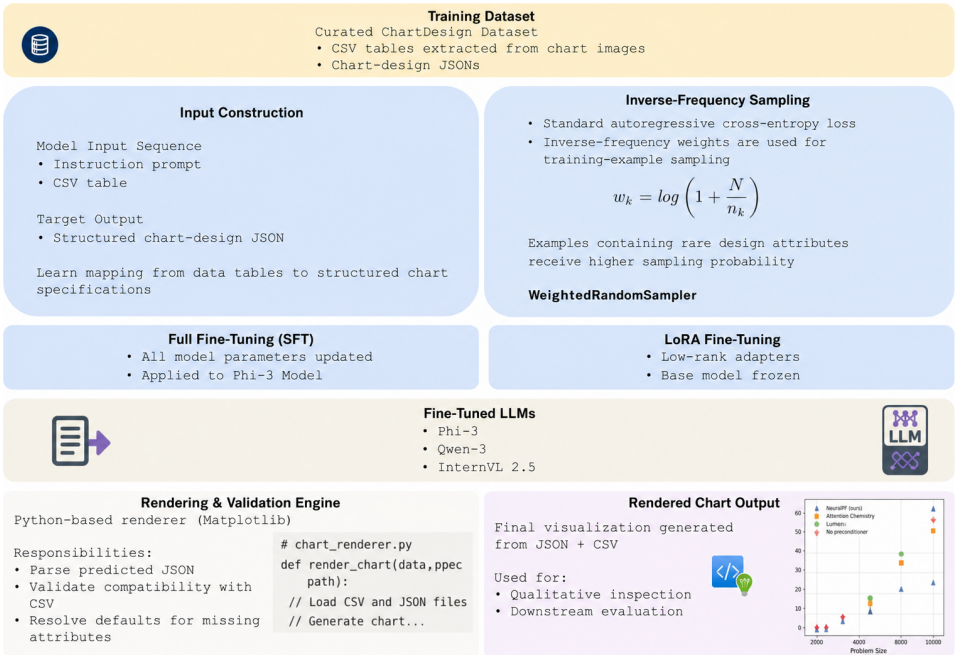

Figure 4: **Post-training pipeline.** The model maps an instruction and CSV table to a structured design JSON using standard autoregressive cross-entropy. Inverse-frequency sampling increases exposure to underrepresented design attributes; adaptation uses either LoRA or full fine-tuning.

and rank-16 adapters are inserted into attention and feed-forward layers. We additionally report full fine-tuning for Phi-3.

To reduce imbalance in design attributes, examples are sampled using inverse-frequency weights. For attribute value k with frequency n_k in a training pool of size N, we define $w_k = \log(1 + N/n_k)$ and combine the applicable attribute weights at the example level. Mini-batches are drawn with PyTorch's `WeightedRandomSampler`. Training uses AdamW for three epochs after excluding the 200 human-verified test charts; the remaining data are divided 90/10 into training and validation splits. Full hyperparameters and prompt formats are given in Appendix A.2.

3.5 Evaluation

Our primary quantitative metric is per-attribute *reference-match accuracy* against the manually verified reference JSON, macro-averaged across evaluated attributes. The metric measures consistency with one human-authored design rather than universal visualization correctness. Exact normalized matches are handled deterministically, while an LLM judge is used for semantic equivalence that is not captured by exact string matching. The judge was validated on 120 human-labelled attribute pairs and achieved 92% pooled agreement with the majority human label. We report this as pooled validation rather than evidence of uniform reliability across every attribute or model.

Reference matching is complemented by rendered-chart evaluation. This is necessary because a model can select a different spacing, legend position, or other stylistic property while still producing a useful visualization. Our human study therefore separates source-design fidelity from readability, appropriateness, visual clarity, information preservation, and pairwise preference. Appendix A.3 gives the evaluation prompt and scoring details.

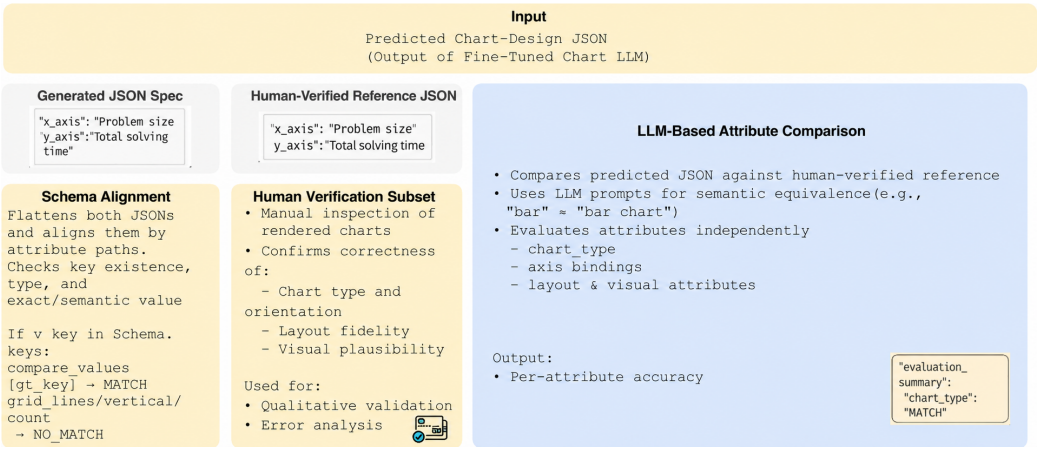

Figure 5: **Reference-match evaluation pipeline.** Predicted and human-verified reference JSONs are aligned by schema field and compared using deterministic normalization together with semantic equivalence judging where needed.

4 Experiments

4.1 Experimental setup

The 200 human-verified charts are excluded from training and validation. The remaining corpus is divided 90/10 for training and validation. We compare matched zero-shot and adapted versions of the primary model backbones, together with a deterministic rule-based visualization recommender and a direct-copy nearest-neighbour baseline. The rule-based system uses table structure, column types, cardinality, and temporal information to choose a chart family and orientation, then fills supported remaining fields with deterministic defaults. The nearest-neighbour baseline retrieves the structurally closest training table and copies its design JSON unchanged. Because semantic fields are not rebound to the test table, this retrieval baseline should be interpreted as a conservative lower-bound test of direct template reuse.

4.2 Main results

Table 1: Reference-match accuracy (%) for the principal controls and adapted backbones on PewResearch and the combined human-verified evaluation set.

Method	Pew	Combined
Rule-based recommender	55.0	53.0
Direct-copy nearest neighbour	61.0	58.0
Qwen-3 zero-shot	52.0	31.0
Phi-3 zero-shot	53.0	32.0
Phi-3 LoRA (combined train)	78.0	72.0
Qwen-3 LoRA (combined train)	84.0	80.0
Phi-3 full FT (combined train)	**88.0**	**82.0**

Table 2: Source-separated reference-match accuracy (%) for the principal comparisons.

Method	Pew	CharXiV	Combined
Rule-based recommender	55.0	51.0	53.0
Direct-copy nearest neighbour	61.0	55.0	58.0
Qwen-3 LoRA (combined train)	84.0	76.0	80.0
Phi-3 full FT (combined train)	**88.0**	**76.0**	**82.0**

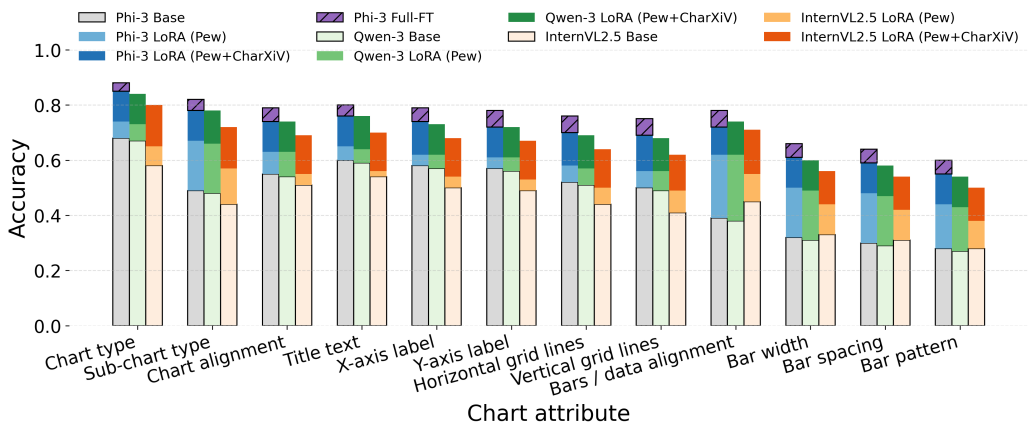

Figure 6: **Attribute-wise reference-match accuracy on the PewResearch + CharXiV evaluation set.** The figure compares base models and adapted configurations across individual design attributes.



Figure 7: **Qualitative comparison across five charts.** Each example shows the source chart, the base Phi-3 output, and the fine-tuned output. Fine-tuning improves major structural and layout decisions while retaining some stylistic differences from the human-authored reference.

Tables 1 and 2 summarize the main quantitative results rather than listing model-by-model scores in prose. Post-training produces large gains over the matched zero-shot backbones and over the two simple control baselines. The source-separated evaluation shows that the gains are not confined to the presentation-oriented PewResearch subset. These comparisons establish a clear benefit from task-specific post-training relative to the evaluated controls, but they do not establish superiority over all modern untuned or proprietary models.

Figure 6 provides an attribute-level view of performance on the combined evaluation set. High-level structural attributes are generally easier than fine-grained properties such as bar width, spacing, and pattern. The same distinction appears in the annotation audit, where structural properties are more consistently labelled than detailed layout attributes.

Figure 7 complements the aggregate scores with rendered examples. Relative to the base Phi-3 model, the fine-tuned model more consistently recovers the source chart's major structural choices, including chart family, orientation, grouping, labels, and layout, while some stylistic differences remain.

4.3 Training-data scaling

We evaluate Qwen-3 LoRA using increasing fractions of the available training split while keeping the evaluation set fixed. Table 3 shows a monotonic improvement over the evaluated range, with the largest gains occurring at smaller data fractions. The result demonstrates that additional supervision

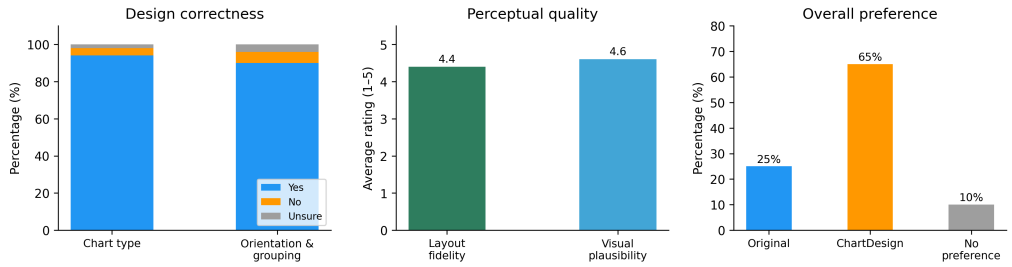

Figure 8: **Human-evaluation summary.** Left: source-design correctness for chart type and orientation/grouping. Center: mean ratings for layout fidelity and visual plausibility. Right: pairwise preference between the original and `ChartDesign` output.

is useful within the current corpus, although it does not by itself rule out source-specific regularities or template similarity.

The scaling curve provides evidence that the post-training gains are not restricted to a single training-set size. Performance improves throughout the evaluated range, with diminishing gains as the complete corpus is approached. We therefore treat scale as one contributor to performance rather than as evidence that the current corpus has saturated the task.

Table 3: Combined reference-match accuracy for Qwen-3 LoRA as training data increases.

Train	10%	25%	50%	75%	100%
Acc. (%)	56.0	65.0	72.0	77.0	80.0

4.4 Human Evaluation

Ten annotators each evaluated 40 chart pairs sampled from the two sources. System identity was hidden and the left/right order was randomized. Five annotators reported prior experience with visualization, data science, or machine learning, while the remaining five had general technical backgrounds. Fleiss' $\kappa = 0.81$ for categorical questions and Krippendorff's $\alpha = 0.82$ for ordinal ratings.

We evaluate three complementary dimensions: source-design fidelity, quality criteria that do not require exact reference matching, and pairwise preference. Figure 8 summarizes the fidelity and preference outcomes, while Table 4 reports the main numerical results compactly.

Table 4: Human-evaluation summary. Fidelity measures compare `ChartDesign` with the source design; quality criteria use a 1–5 scale. Pairwise preference has a remaining 10% "neither" response. Ratings are descriptive because observations are repeated across annotators and charts.

Measure	**Original**	**`ChartDesign`**
Chart-type agreement	–	94%
Orientation/grouping agreement	–	90%
Layout fidelity	–	4.4
Visual plausibility	–	4.6
Readability	4.6	4.5
Design appropriateness	4.6	4.4
Visual clarity	4.7	4.5
Information preservation	4.7	4.6
Overall design quality	4.7	4.5
Pairwise preference	25%	65%

The absolute quality ratings and pairwise preference should not be treated as interchangeable statistics. Original charts receive slightly higher mean ratings on the individual quality criteria, while `ChartDesign` is selected more often in direct comparisons. These measurements answer different

questions, so we report both without collapsing them into a single notion of design quality. The full questionnaire, repeated-measures caveat, additional adapted-model configurations, and reproducibility details are provided in Appendix A.10 and the surrounding supplementary sections.

4.5 Interpretation

The experiments support two conclusions. First, task-specific post-training materially increases agreement with human-authored chart specifications relative to the evaluated zero-shot, rule-based, and direct-copy controls. Second, reference agreement and visualization quality are distinct. We therefore separate three layers of evidence: data and encoding validity, readability and task appropriateness, and stylistic agreement with a reference. A reference mismatch in spacing, grid-line density, or legend placement may be benign, whereas a wrong encoding or misleading representation of the supplied data is a more consequential error. This distinction motivates the human evaluation and the limitations discussed below.

5 Limitations

The current study has several limitations. The training-annotation audit is pooled across the corpus, and the reported numeric extraction errors are conditional on recoverable cells that can be aligned to corrected CSVs. The semantic-judge validation is likewise pooled over 120 attribute pairs rather than stratified by every attribute and model. Our baseline suite includes simple rule-based and retrieval controls but not a frontier proprietary structured-output model or a substantially larger open-weight model, and the retrieval baseline does not adapt data-dependent fields after retrieval. The human study contains repeated judgments over 40 charts, so the reported ratings are descriptive rather than an analysis that treats all judgments as independent. Finally, the design schema is library-neutral but does not guarantee pixel-identical rendering across plotting systems. Appendix A.13 discusses these limitations in more detail.

6 Conclusion

We introduced `ChartDesign`, a framework for predicting structured chart-design specifications from tabular data. The study combines automatically generated training pairs, a human-verified evaluation set, compact-model post-training, annotation auditing, and human evaluation that separates source-design fidelity from visualization quality. Post-training yields substantial gains over the evaluated rule-based, direct-copy retrieval, and matched zero-shot controls, while the human study shows that generated charts can remain visually plausible even when they differ from the source design. These findings support learning recurring human-authored design preferences through a renderer-neutral representation, while also emphasizing that chart design is underdetermined and should not be evaluated through reference agreement alone.

AI use statement

Generative AI tools were used for language editing, LaTeX formatting, and literature-search assistance. All AI-assisted material was reviewed and edited by the authors. A vision-language model (Phi-3.5-Vision Instruct) and a language-model judge are also components of the experimental methodology; their roles are described in Sections 3 and A.3. The authors take responsibility for the final content, claims, analyses, and artifacts in this work.

Ethics statement

This work uses publicly available chart datasets and a human evaluation of chart quality. The study collects no sensitive personal information. The main risks are evaluation bias from model-based judging and subjectivity in visualization preferences; we therefore use a human-verified test set, report human agreement measures, and distinguish reference agreement from design quality.

Reproducibility statement

Dataset construction, training, evaluation, prompts, compute, and the human-study protocol are documented in Sections 3–4 and Appendix A. The supplementary material includes a machine-readable design schema in `schema/chartdesign.schema.json`. Upon publication, the authors plan to release processing scripts, train/validation/test splits, evaluation prompts, and LoRA adapters, subject to the source datasets' licences.

References

William S. Cleveland and Robert McGill. Graphical perception and graphical methods for analyzing scientific data. *Science*, 229(4716):828–833, 1985.

Victor Dibia and Çagatay Demiralp. Data2vis: Automatic generation of data visualizations using sequence-to-sequence neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 2019.

Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. DVQA: Understanding data visualizations via question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5648–5656, 2018. doi: 10.1109/CVPR.2018.00592.

Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Ákos Kádár, Adam Trischler, and Yoshua Bengio. FigureQA: An annotated figure dataset for visual reasoning. In *ICLR Workshop on Learning to Generate and Understand*, 2018. URL `https://arxiv.org/abs/1710.07300`.

Woosung Koh, Janghan Yoon, MinHyung Lee, Youngjin Song, Jaegwan Cho, Jaehyun Kang, Taehyeon Kim, Se-Young Yun, Youngjae Yu, and Bongshin Lee. C^2: Scalable Auto-Feedback for LLM-based Chart Generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4525–4566, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.naacl-long.232. URL `https://aclanthology.org/2025.naacl-long.232/`.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Junyu Luo, Zekun Li, Jinpeng Wang, and Chin-Yew Lin. Chartocr: Data extraction from charts images via a deep hybrid framework. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1917–1925, 2021.

Jock Mackinlay, Pat Hanrahan, and Chris Stolte. Show me: Automatic presentation for visual analysis. *IEEE transactions on visualization and computer graphics*, 13(6):1137–1144, 2007.

Jock D. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, 5(2):110–141, 1986.

Paula Maddigan and Teo Susnjak. Chat2vis: Generating data visualizations via natural language using chatgpt, codex and gpt-3 large language models. *Ieee Access*, 11:45181–45193, 2023.

Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the association for computational linguistics: ACL 2022*, pp. 2263–2279, 2022.

Pew Research Center. Pew research center visualizations. `https://www.pewresearch.org/`, 2015. Accessed 2025-10-26.

Steven F Roth, John Kolojejchick, Joe Mattis, and Jade Goldstein. Interactive graphic design using automatic presentation knowledge. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 112–117, 1994.

Benny Tang, Angie Boggust, and Arvind Satyanarayan. VisText: A benchmark for semantically rich chart captioning. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7268–7298, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.401. URL https://aclanthology.org/2023.acl-long.401/.

Yuan Tian, Weiwei Cui, Dazhen Deng, Xinjing Yi, Yurun Yang, Haidong Zhang, and Yingcai Wu. Chartgpt: Leveraging llms to generate charts from abstract natural language. *IEEE Transactions on Visualization and Computer Graphics*, 31(3):1731–1745, 2024.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, Alexis Chevalier, Sanjeev Arora, and Danqi Chen. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. *arXiv preprint arXiv:2406.18521*, 2024.

Kanit Wongsuphasawat, Leilani Battle, Arvind Srinivasan, et al. Voyager: Exploratory data analysis via recommendation. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 2016.

Songheng Zhang, Haotian Li, Huamin Qu, and Yong Wang. Adavis: Adaptive and explainable visualization recommendation for tabular data. *IEEE Transactions on Visualization and Computer Graphics*, 30(9):5923–5938, 2024. doi: 10.1109/TVCG.2023.3316469.

A Appendix

A.1 Prompt templates

CSV extraction prompt.

> You are a chart understanding model. Carefully read the following chart image. Extract the data shown in the chart as CSV format. If the image contains multiple charts, output each separately. Include column headers. Output only CSV with no additional text.

Design annotation prompt.

> You are a vision-language model that analyzes chart images and their associated CSV data. Given the chart image and CSV data, generate a JSON object describing the chart structure and layout. Follow the specified schema. Only include keys that are relevant for the given chart type; omit non-applicable fields.

Evaluation prompt.

> You are evaluating agreement between predicted chart design attributes and a human-verified reference. For each pair of reference and predicted values, respond with MATCH if they are semantically equivalent and NO_MATCH otherwise. Use natural-language reasoning to decide semantic equivalence.

A.2 Training details

The 200 human-verified evaluation charts are excluded before any model adaptation. The remaining 1,918 examples are divided into training and validation sets with a 90/10 split. Phi-3 and Qwen-3 LoRA experiments use rank-16 adapters inserted into attention and feed-forward layers, while full fine-tuning updates all Phi-3 parameters. Models are trained for three epochs with AdamW at learning rate 10^{-4} using mixed-precision training. Maximum sequence lengths are 14k tokens for Phi-3 and 8k for Qwen. Mini-batches contain four examples and are drawn with replacement according to the inverse-frequency sampling weights described in Section 3.

The inverse-frequency weight for an attribute value k with training frequency n_k is

$$w_k = \log\left(1 + \frac{N}{n_k}\right),$$

where N is the number of training examples. Example-level sampling weights combine the weights of applicable values, so underrepresented design choices are encountered more frequently during optimization. The loss itself remains standard autoregressive cross-entropy; the class balancing is implemented through sampling rather than token-level loss reweighting.

A.3 Evaluation details

Predicted and reference JSONs are flattened into attribute paths before scoring. Exact normalized matches are evaluated deterministically. The LLM judge is used only for semantic equivalence that cannot be resolved by exact normalization. Each applicable attribute is scored as MATCH or NO_MATCH, and attribute-wise accuracies are macro-averaged to obtain the headline reference-match score. Because applicable attributes vary by chart family, reference-match accuracy is interpreted as consistency with the human-verified source specification rather than as universal correctness.

The judge was checked on 120 human-labelled attribute pairs and agreed with the majority human decision in 92% of the pooled sample. This validation is not presented as attribute- or model-stratified reliability. The human-verified evaluation set is separate from the automatically labelled training examples, so annotation noise in the training supervision does not directly determine the reference labels used for scoring.

A.4 Comparison with related systems

Table 5: Representative visualization systems and datasets. "Structured information" summarizes the representations available to the method or dataset, not only its final task output.

System	Structured information	Primary task	Evaluation
Data2Vis Dibia & Demiralp (2019)	Table / Vega-Lite	Specification generation	Specification metrics
AdaVis Zhang et al. (2024)	Table / candidate visualizations	Recommendation	Recommendation / user study
ChartGPT Tian et al. (2024)	Table / plot code	Chart generation	User evaluation
C^2 Koh et al. (2025)	Plot code / rendered chart	Auto-feedback refinement	Human preference
VisText Tang et al. (2023)	Image / table / scene graph / Vega-Lite	Captioning	NLG metrics
`ChartDesign`	CSV / design JSON	Design-specification prediction	Reference match / human quality

A.5 Corpus and schema summary

The corpus contains 1,101 PewResearch charts and 1,017 CharXiV charts. One hundred charts from each source are reserved for the manually verified evaluation set, leaving 1,918 charts for training and validation. The corpus covers bar, line, area, scatter, pie, box, and histogram families, including grouped and stacked bar variants. We report the verified source totals used throughout this submission together with schema-domain diversity. Per-family source counts are not used in the quantitative claims of this paper.

Table 6: Corpus totals and observed schema-domain diversity used in this submission.

Statistic	PewResearch	CharXiV
Source charts	1,101	1,017
Human-verified evaluation charts	100	100
Training/validation pool	1,001	917
Alignment values	3	3
Horizontal grid-line values	0–4	0–6
Vertical grid-line values	0–1	0–4
Bar-width values	5	4
Bar-spacing values	4	5
Style patterns	3	3

A.6 Training-annotation quality audit

The audit covers 300 randomly sampled training examples that are separate from the held-out evaluation set. For each example, the extracted CSV and design JSON are compared with the source

chart. The extraction target is restricted to values or summary statistics visibly encoded in the published chart; unavailable latent raw observations are not treated as extraction targets.

Numeric NMAE and NRMSE are computed only for recoverable numeric cells that can be aligned between the automatic extraction and the human-corrected CSV. Missing or unmatched rows, series, and structurally incorrect tables are not silently converted into numeric cell errors; the reported numeric metrics therefore characterize aligned-cell fidelity rather than all possible extraction failures.

Table 7: CSV extraction quality on the 300-chart audit.

Metric	Result
CSV structure accuracy	96.7%
CSV value agreement	94.1%
Normalized MAE	0.034
Normalized RMSE	0.057

Table 8: Design-attribute agreement and inter-annotator reliability on the training audit.

Attribute	Agreement (%)	Krippendorff's α
Chart type	97.7	0.95
Sub-chart type	93.3	0.87
Orientation	96.0	0.93
Text elements	91.0	0.82
Legend attributes	92.7	0.84
Grid-line attributes	89.0	0.78
Width and spacing	86.7	0.73
Style pattern	90.3	0.80
Overall design attributes	**92.1**	**0.84**

The audit is pooled across both sources, so it should not be interpreted as establishing identical annotation quality for PewResearch, CharXiV, or every rare chart family. Source- and family-stratified auditing would provide a more granular characterization of supervision quality.

A.7 Design schema and renderer portability

The top-level `chart_type` supports `bar`, `line`, `area`, `scatter`, `pie`, `boxplot`, and `histogram`. Additional objects encode chart subtype, orientation, text elements, axes, legend properties, grid lines, mark grouping, marker style, size and spacing, and box-plot-specific options. Fields that are not applicable to a chart family are omitted. The accompanying `schema/chartdesign.schema.json` records documented field types, enumerations, numeric constraints, and family-applicability notes.

The schema is informed by visualization principles concerning graphical encoding, orientation, visual search, grid-line density, mark discriminability, and semantic context Cleveland & McGill (1985); Mackinlay (1986). We use these principles to motivate the represented design factors, not to claim that agreement with a single reference establishes perceptual optimality.

Renderer independence means that the specification is not tied to one plotting-library syntax. It does not guarantee pixel-identical output because libraries can apply different defaults or expose different controls. Figure 2 in the main paper therefore demonstrates portability rather than measured cross-renderer property preservation.

A.8 Full matched-backbone and adaptation results

The primary paper emphasizes matched comparisons between the backbones used for adaptation. For completeness, Table 10 also reports the identifiable adapted-model configurations evaluated under Pew-only and combined-corpus training.

Table 9: Semantic mapping from schema fields to renderer-level intent. This table documents intended portability, not measured cross-renderer preservation rates.

Schema field	Renderer-level intent
`chart_type`, `sub_chart_type`	chart primitive / composition
`chart_alignment`	horizontal or vertical orientation
`text_elements`, `axes`	titles, labels, categories, ranges
`legend`	visibility, position, labels / frame
`grid_lines`	horizontal / vertical grids
`bars_or_data_points`	grouping, markers, mark style
`size_and_spacing`	mark dimensions and spacing
`boxplot_style`	whiskers, fliers, mean-marker options

Table 10: Reference-match accuracy (%) for the retained, identifiable model configurations.

Method	Pew	Combined
Rule-based recommender	55.0	53.0
Direct-copy nearest neighbour	61.0	58.0
Qwen-3 zero-shot	52.0	31.0
Phi-3 zero-shot	53.0	32.0
Phi-3 LoRA (Pew train)	74.0	60.0
Qwen-3 LoRA (Pew train)	72.0	61.0
InternVL2.5 LoRA (Pew train)	75.0	62.0
Phi-3 LoRA (combined train)	78.0	72.0
Qwen-3 LoRA (combined train)	84.0	80.0
InternVL2.5 LoRA (combined train)	80.0	70.0
Phi-3 full FT (combined train)	**88.0**	**82.0**

A.9 Baseline details

The rule-based recommender chooses chart family and orientation from table structure, column types, cardinality, and temporal information, then fills supported remaining fields with deterministic defaults. It is a custom deterministic control rather than an implementation of Data2Vis or AdaVis.

The direct-copy nearest-neighbour baseline represents each table through row and column counts, column data types, counts of categorical and numerical columns, temporal-field presence, missing-value statistics, and numerical summaries. It retrieves the structurally closest training table and reuses its design JSON unchanged. Because semantic fields are not rebound to the test table, this baseline is intentionally a conservative lower bound on retrieval-based template reuse rather than an adaptive template-transfer system.

A.10 Human-study protocol and results

Ten annotators each judged 40 original/generated chart pairs sampled from PewResearch and CharXiV, yielding repeated observations over the same set of charts and raters. The examples covered bar, line, scatter, area, box, and pie charts. System identity was hidden, and the left/right order of original and generated charts was randomized for every comparison. Five annotators reported prior experience with visualization, data science, or machine learning, while the remaining five had general technical backgrounds.

Annotators answered the following questions:

1. **Chart type correctness:** Does the predicted chart use the same chart family (bar, line, scatter, area, box, or pie) as the original? Responses: Yes / No / Unsure.
2. **Orientation and grouping:** Are the orientation and grouping or stacking of elements consistent with the original? Responses: Yes / No / Unsure.
3. **Layout fidelity:** How closely does the predicted chart match the original layout, including titles, axis labels, legend placement, and spacing? Likert scale: 1–5.

4. **Visual plausibility:** Does the predicted chart look visually plausible and free of obvious rendering errors such as overlaps or truncation? Likert scale: 1–5.
5. **Readability:** How easy is the chart to read and interpret? Likert scale: 1–5.
6. **Design appropriateness:** How appropriate are the selected visual encodings for the underlying data? Likert scale: 1–5.
7. **Visual clarity:** How clearly does the chart present important patterns in the data? Likert scale: 1–5.
8. **Information preservation:** How well does the visualization preserve and communicate the relationships present in the data? Likert scale: 1–5.
9. **Overall design quality:** How would you rate the overall quality of the visualization? Likert scale: 1–5.
10. **Overall preference:** Which chart better communicates the main information or patterns of the data? Responses: Original / Predicted / Both equally.

The first four questions measure fidelity to the source design. Questions 5–9 assess visualization quality without requiring exact attribute matching, although ratings were collected in the paired interface. Overall inter-annotator agreement is substantial, with Fleiss' $\kappa = 0.81$ for categorical questions and Krippendorff's $\alpha = 0.82$ for ordinal ratings.

Table 11: Aggregated source-fidelity and preference results from the human study.

Type	Orient.	Layout	Visual	Prefer original	Prefer `ChartDesign`
94%	90%	4.4/5	4.6/5	25%	65%

The remaining 10% of pairwise judgments indicate no preference. Because judgments are repeated across both annotators and charts, these results are presented descriptively rather than treating all pair evaluations as independent samples.

Table 12: Human ratings on quality criteria that do not require exact reference matching. Scores use a 1–5 scale.

Method	Read.	Approp.	Clarity	Info.	Overall
Original	4.6	4.6	4.7	4.7	4.7
`ChartDesign`	4.5	4.4	4.5	4.6	4.5

The absolute ratings and pairwise preferences answer different questions. The original charts receive slightly higher mean absolute quality ratings, whereas `ChartDesign` is chosen more frequently in pairwise preference. We therefore report both outcomes without treating either statistic as a universal ranking of design quality.

A.11 Mismatch interpretation

Macro-averaged attribute accuracy cannot be converted directly into a percentage of failed charts. We therefore do not report a failure-rate denominator derived from one minus the aggregate score. Observed reference mismatches include grouped-versus-stacked subtype differences, bar-spacing differences, grid-line count differences, legend ordering or placement, and other fine-grained style differences. These mismatches should be interpreted at different levels: data or encoding errors can compromise correctness, readability or task-appropriateness problems can impair interpretation, and stylistic differences can be harmless alternatives. A focused human assessment of disagreement cases would be required to determine which reference mismatches materially reduce visualization quality.

A.12 Reproducibility, release plan, and compute

The authors plan to release the CSV/design-JSON pairs with train/validation/test splits, processing and annotation scripts, inference and evaluation code including the judge prompt, and LoRA adapter

weights for the primary fine-tuned models, subject to the source datasets' licences. Code is intended for release under the Apache 2.0 licence; source chart images remain subject to the licences and redistribution terms of PewResearch and CharXiV.

Table 13: Recorded compute and inference requirements. Training experiments use a single NVIDIA RTX A6000 with 48 GB VRAM.

Task	Recorded compute / latency
Phi-3 LoRA, combined corpus	approximately 6 GPU-hours
Phi-3 full fine-tuning, combined corpus	approximately 24 GPU-hours
Inference per chart	less than 3 seconds
Three-shot prompting per chart	approximately 8–12 seconds

A.13 Full limitations

Chart coverage. The schema focuses on common static 2D chart families: bar, line, scatter, area, pie, box, and histogram charts. Maps, network diagrams, interactive dashboards, and coordinated multi-view systems are outside the current scope.

Automatically generated supervision. Most training CSV and design annotations are generated by a vision-language model rather than exhaustively labelled by humans. The 300-chart audit provides evidence about the resulting supervision quality, but it is pooled rather than stratified by source and rare chart family, and the numeric extraction metrics are conditional on recoverable aligned cells rather than structural failures. A controlled comparison of automatic versus corrected training labels would better isolate the effect of annotation noise.

Reference-based evaluation. A data table does not uniquely determine a single ideal visualization. User intent, audience, and communication goals can change which design is most appropriate. Reference-match accuracy therefore measures consistency with one human-authored design rather than universal correctness. A focused assessment of disagreement cases, multiple acceptable references, or intent-conditioned evaluation would provide stronger evidence about which mismatches matter for interpretation.

LLM judge. Judge validation is pooled over 120 attribute pairs. Complete reproducibility would require a versioned judge checkpoint, deterministic numeric tolerances, explicit malformed-JSON and missing/extra-key rules, protection against missing fields escaping penalties after an incorrect chart-family prediction, and attribute- and model-stratified validation.

Baseline coverage. The current baselines include matched zero-shot backbones, a simple rule-based recommender, and direct-copy retrieval. They do not include a frontier proprietary structured-output model or a substantially larger open-weight model. The retrieval baseline also does not adapt data-dependent fields after retrieval. Stronger untuned models, validation-tuned prompting, and adaptive template transfer remain important comparisons.

Human evaluation. The study contains ten annotators and forty chart pairs with repeated observations across both raters and charts. The current paper therefore reports the human outcomes descriptively rather than presenting an analysis that treats all 400 judgments as independent. A dependence-aware model with chart and annotator effects, per-question agreement, a fully identified evaluated checkpoint, and explicit rendering details would strengthen inference about method differences.

Rare types and memorization. Rare-family generalization is not separately quantified in the current evidence because a rigorous analysis requires exact family counts, per-family uncertainty, and a source- or template-grouped split to test near-template memorization.

Robustness and masking. Robustness to formatting changes, information removal, and numerical corruption is not comprehensively evaluated here. Controlled formatting-invariance tests, multiple

corruption levels, explicit invariant target attributes, and full/metadata-only/data-only controls would be required for stronger conclusions.

Renderer portability. The design JSON is renderer-neutral at the representation level, but this does not guarantee exact property preservation across plotting libraries. Measured rates of exactly preserved, approximately preserved, and unsupported properties would be needed to quantify cross-renderer fidelity.