Supplementary Material to:

From Figures to Datasets: Vision–Language Models for Quantitative Data Extraction

Léa C. S. Gabay[1], Daniel P. Costa[1], Pedro S. F. Mendes[1], Catarina Barata[2]

[1] Chemical Engineering Department & Centro de Química Estrutural – Institute of Molecular Sciences, Instituto Superior Técnico, Lisboa, Portugal

[2] Electrical and Computer Engineering Department & Institute for Systems and Robotics (ISR/IST), LARSyS, Instituto Superior Técnico, Lisboa, Portugal

June, 2026

Corresponding author: Catarina Barata (e-mail:ana.c.fidalgo.barata@tecnico.ulisboa.pt)

This supplementary section provides additional experimental details, extended analyses, and illustrative examples supporting the results presented in the main manuscript. Specifically, we detail the evaluation thresholds, dataset size selection procedure, comparative characteristics of synthetic and literature-derived figures, extended benchmark analysis, and qualitative baseline model outputs.

A . Testing Parameters

This section provides a detailed description of the evaluation settings used throughout the experimental analysis, including similarity thresholds and the determination of test set size.

A.1 . Thresholds

Model outputs were evaluated using both textual and numerical similarity criteria, as described in Section 4.3.1 of the main manuscript. Here, we provide additional justification and sensitivity analysis supporting the chosen thresholds.

Textual similarity between predicted and ground-truth labels (e.g., axes titles, legend entries, series names) was computed using the Levenshtein ratio. A threshold of $\mathbf{T_{text}} = \mathbf{0.85}$ was selected. Figure 1 illustrates model performance as a function of the text similarity threshold. As shown, values below 0.80 introduce excessive tolerance, while values above 0.95 become overly restrictive and penalize minor formatting variations.

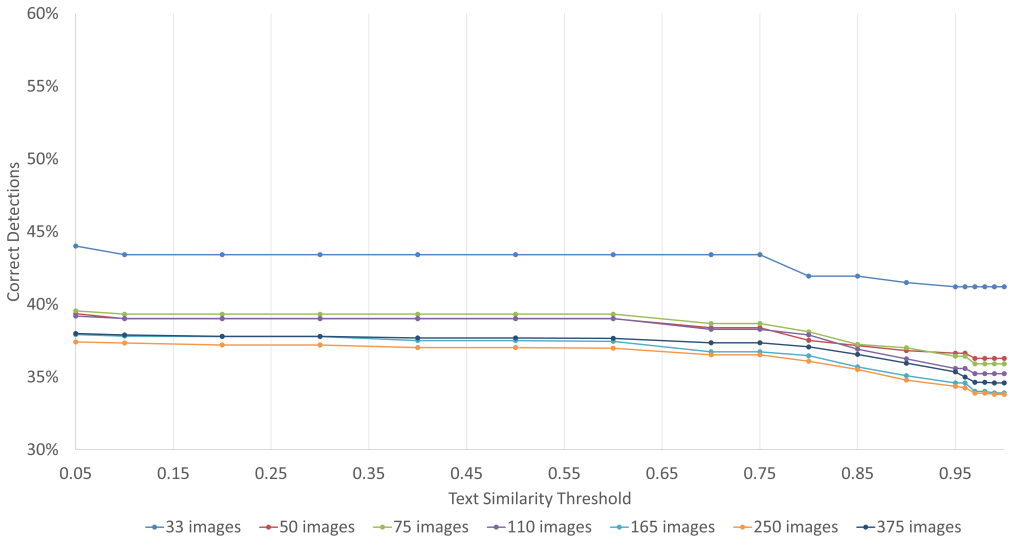


Figure 1: Sensitivity analysis of point-detection accuracy as a function of the textual similarity threshold (T_{text}). Performance stabilizes around 0.85, which balances tolerance to minor formatting variations with strict semantic matching.

Numerical agreement between predicted and reference data points was assessed using normalized Euclidean distance. A tolerance threshold of $\mathbf{T_{num}} = \mathbf{0.1}$ (10%) was adopted. The selected value balances robustness to minor extraction noise with the need for quantitative precision consistent with experimental uncertainty in catalysis studies.

A.2 . Number of Images for Test

To determine an appropriate number of evaluation images, we conducted a stabilization analysis. Performance metrics were computed incrementally as a function of increasing test-set size.

Figure 2 shows that model performance stabilizes within the range of 50–200 images, beyond which variance reduction becomes marginal. Below 100 images, metric fluctuations are significant, indicating insufficient statistical representativeness.

Based on this analysis, all reported evaluations were conducted using test sets within the 150–200 image range.

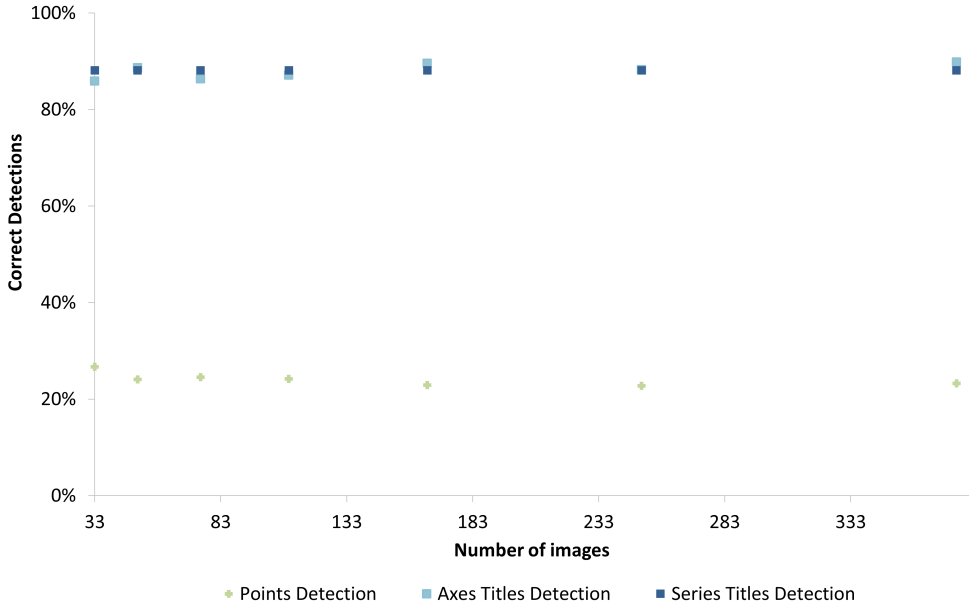


Figure 2: Stabilization analysis of evaluation performance as a function of test-set size. Performance variance decreases and stabilizes within the range of 150–200 images.

B . Literature Charts vs Synthetic Charts

To ensure that the synthetic dataset realistically reflects real-world catalytic figures (Figure 3), we performed a comparative structural analysis between literature-derived and synthetic charts.

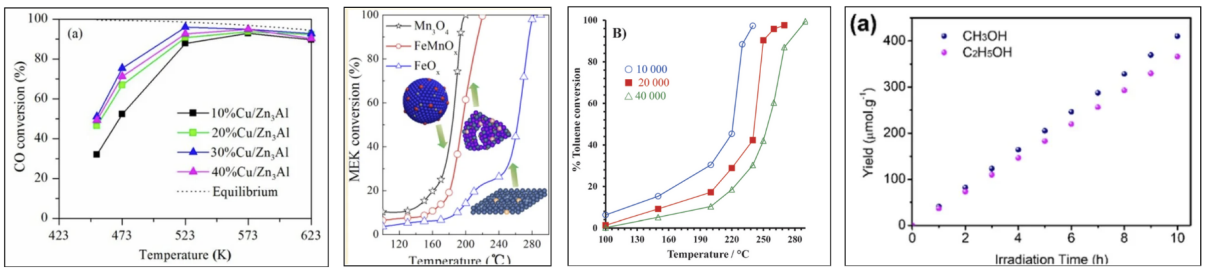


Figure 3: Representative examples of target figures.

Key characteristics examined include:

- Number of points per series
- Number of series per figure
- Marker types and colour diversity
- Presence of trend lines and fitted equations
- Legend complexity and positioning
- Axis formatting (linear vs logarithmic)
- Presence of subplots or embedded graphical elements

1. Distribution Fitting

To enable statistically grounded synthetic data generation, probability distributions were fitted to each numerical feature. Best-fit models were selected based on minimization of the sum of squared errors (SSE).

Table 1 summarizes the selected distributions and their estimated parameters.

Several features exhibit log-normal behavior, particularly those related to point counts and total data density, indicating asymmetric distributions with heavy upper tails. Overlap percentages are better modeled using a Gamma distribution, reflecting their strictly positive and skewed nature.

These fitted distributions were subsequently used to parameterize the synthetic chart generation pipeline, ensuring statistical alignment with real data.

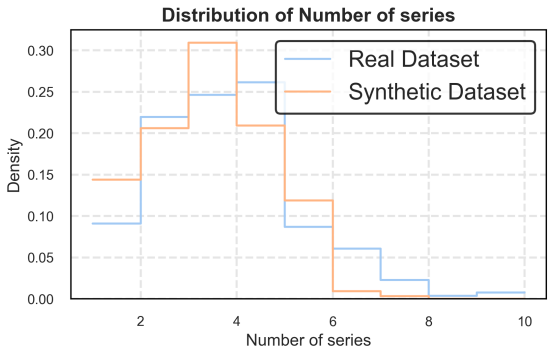
Table 1: Best-fit probability distributions and parameter estimates for structural features extracted from literature figures.

Feature	Best-Fit Distribution	Parameters	SSE
Number of Series	Normal	$\mu = 3.39,\ \sigma = 1.56$	0.746
Max Points per Series	Log-Normal	$s = 0.562,\ \text{loc} = 0.196,\ \text{scale} = 7.81$	0.00909
Total Points	Log-Normal	$s = 0.788,\ \text{loc} = 0.178,\ \text{scale} = 20.7$	7.37893×10^{-5}
Percentage of Overlap	Gamma	$a = 0.447,\ \text{scale} = 15.9$	0.00877
Original Width	Normal	$\mu = 600,\ \sigma = 146$	5.402×10^{-6}
Original Height	Log-Normal	$s \approx 0$	6.156×10^{-6}
Resolution	Normal	$\mu = 2.54 \times 10^5,\ \sigma = 8.57 \times 10^4$	7.587×10^{-12}
Ratio	Log-Normal	$s = 0.168,\ \text{scale} = 1.53$	1.994798862
Minimum Dimension	Normal	$\mu = 455,\ \sigma = 84.8$	4.817×10^{-6}

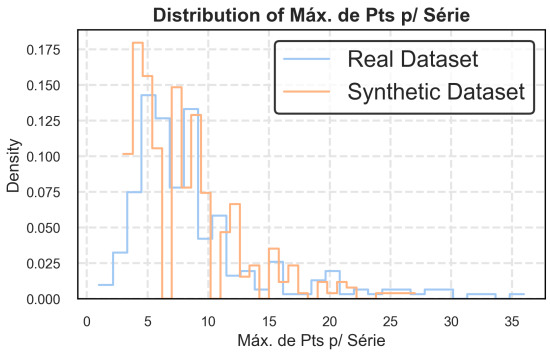
2. Comparison Between Literature and Synthetic Distributions

Figure 4 compares the structural characteristics of literature-derived figures and synthetically generated images.

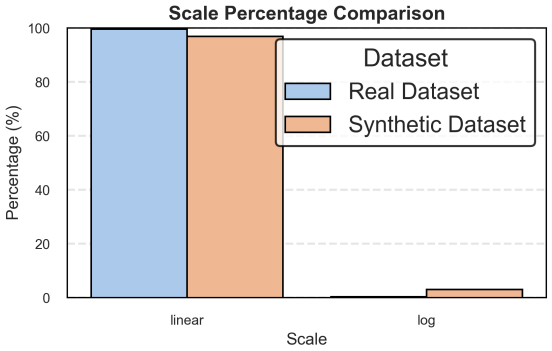
The synthetic dataset closely reproduces the empirical distributions observed in real data across key structural dimensions, including series count, point density, scale usage, and legend complexity. Minor deviations are primarily attributable to generation constraints and controlled parameter bounds.



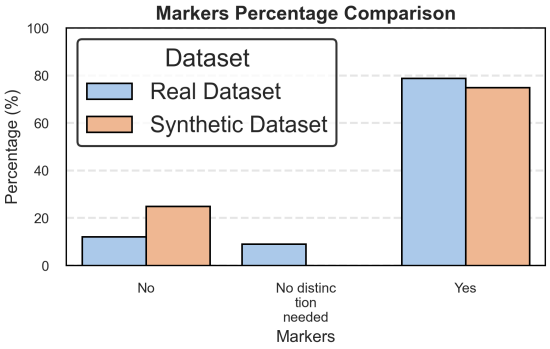
(a) Number of series distribution.



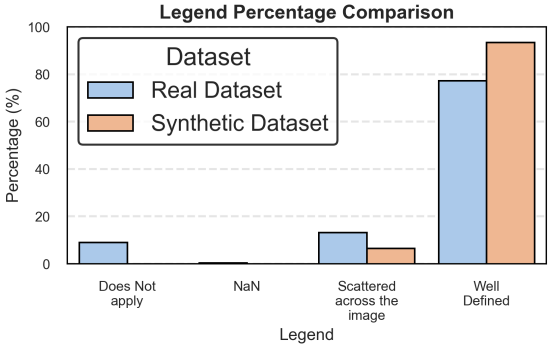
(b) Maximum number of points per serie distribution.



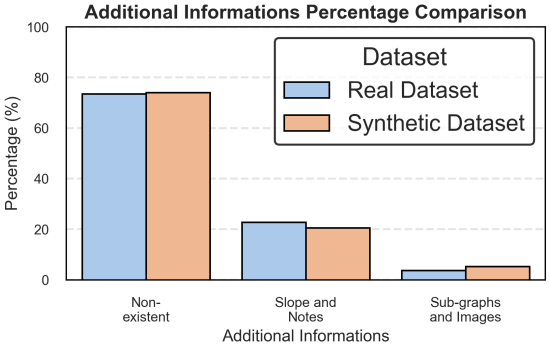
(c) Scale type count



(d) Markers usage.



(e) Legend description distribution.



(f) Additional information in the Plot Area Distribution.

Figure 4: Comparison of structural distributions between literature-derived and synthetic figures. The synthetic dataset reproduces key statistical characteristics observed in real-world charts.

3.Generation Image Parameters

Table 2 summarizes the configuration settings used to generate the three synthetic scatter-plot datasets described in this work. The table provides a side-by-side comparison of all generation parameters, including figure dimensions,

sampling distributions, visual styling options, annotation probabilities, and data modelling strategies.

Examples 1 and 2 rely on uniform sampling strategies with controlled randomness, while Example 3 incorporates literature-inspired statistical distributions (truncated normal and log-normal) to better approximate realistic scientific data variability. All datasets share common visualization primitives (markers, line styles, legend handling, and axis label sourcing) but differ in overlap tolerance, annotation frequency, and structural complexity.

Table 2: Unified Configuration Comparison of Synthetic Dataset Examples

Parameter	**Description**	**Ex. 1**	**Ex. 2**	**Ex. 3**
Function Name	Data generation function	generate trend points rescaled	generate trend points	generate trend points3
Number of Images	Total generated figures	1820	1820	1820
Width (in)	Figure width range	10–15	10–15	10–15
Height Ratio	Height/width ratio	0.6–1.7	0.6–1.7	0.699–3.2
Series (min–max)	Number of data series	2–10	2–10	1–9
Points (min–max)	Points per series	5–30	5–55	3–35
Marker Sizes	Available marker sizes	80–260	80–260	80–260
Font Sizes	Available font sizes	14–24	14–24	12–24
Max Overlap	Allowed overlap (%)	40%	60%	40%
Overlap Mode	Alignment probability	Bern $p = 0.70$	Uniform	Bern $p = 0.70$
Error Bars	Probability enabled	Bern $p = 0.15$	Uniform	15%
Grid	Grid probability	Bern $p = 0.05$	Bern $p = 0.05$	Bern $p = 0.1$
Vertical Lines	Add reference lines	—	—	Bern $p = 0.20$
Notes	Add annotations	$p = 0.1$	Bern $p = 0.1$	Bern $p = 0.25$
Rectangle	Add center rectangle	—	—	Bern $p = 0.2$
Legend	Legend probability	Always	Always=	Bern $p = 0.90$

Parameter	**Description**	**Ex. 1**	**Ex. 2**	**Ex. 3**
Trend Types	Available models	Polynomial	Lin/Quad/Cub/ Log/Spline	Lin/Quad/Cub/ Log/Spline
Series Dist.	Distribution (series)	Uniform	Uniform	Trunc. Normal
Points Dist.	Distribution (points)	Uniform	Uniform	Trunc. Log-Norm
Sub-image	Embed object image	—	—	Bern $p = 0.2$

C . Benchmark Analysis

This section presents extended benchmark analyses complementing Section 2.2.2 of the main manuscript. In addition to reporting absolute benchmark scores (OCRBench v2, Chart-to-Table, Chart-to-Text, CharXiv, and ChartQA), we evaluate the extent to which performance on these benchmarks correlates with our downstream structured extraction metrics.

Figure 8 illustrates cross-benchmark correlation analyses, reporting coefficients of determination (R^2) and the number of evaluated models (n) for each comparison.

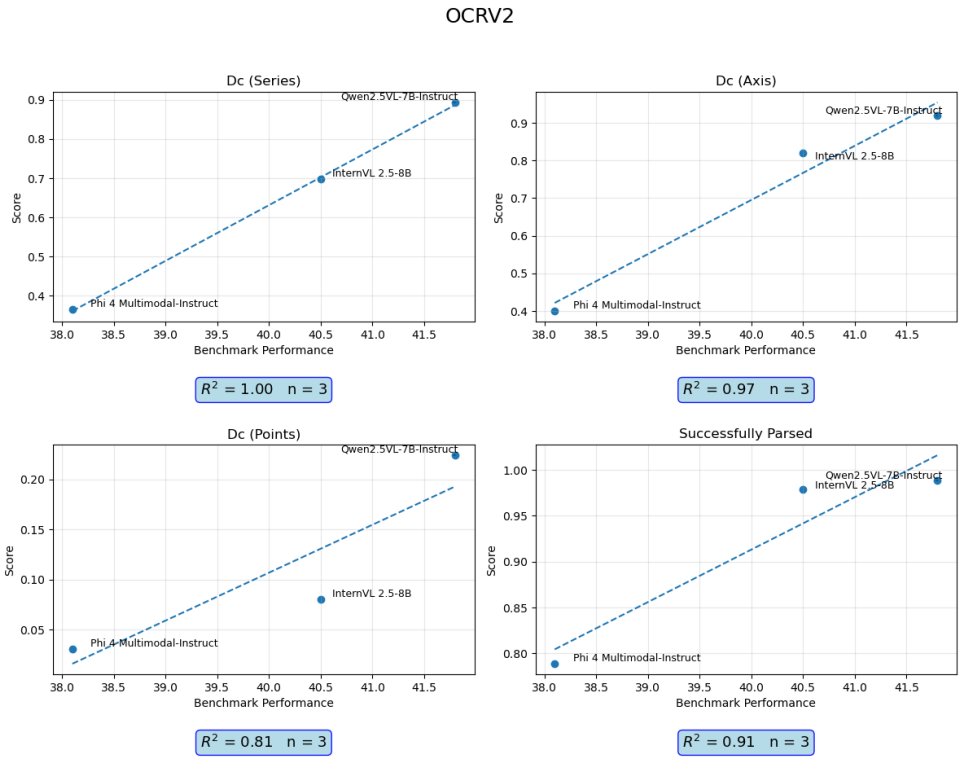


Figure 5: Correlation between OCRBench v2 scores and downstream structured extraction performance.

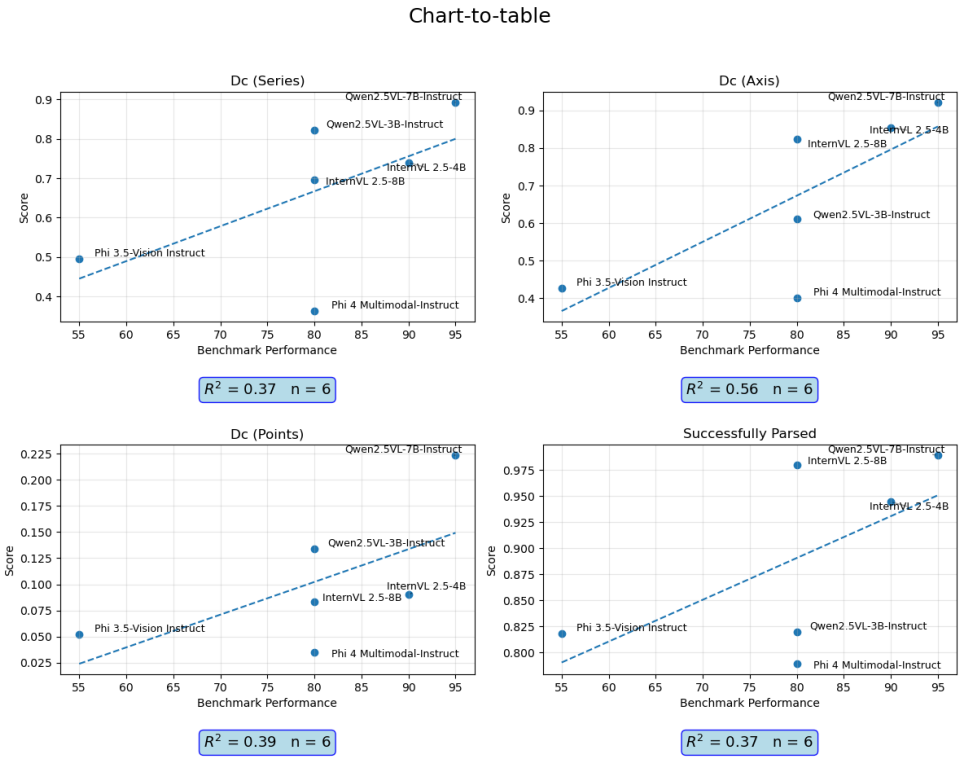


Figure 6: Correlation between Chart-to-Table benchmark scores and downstream structured extraction performance.

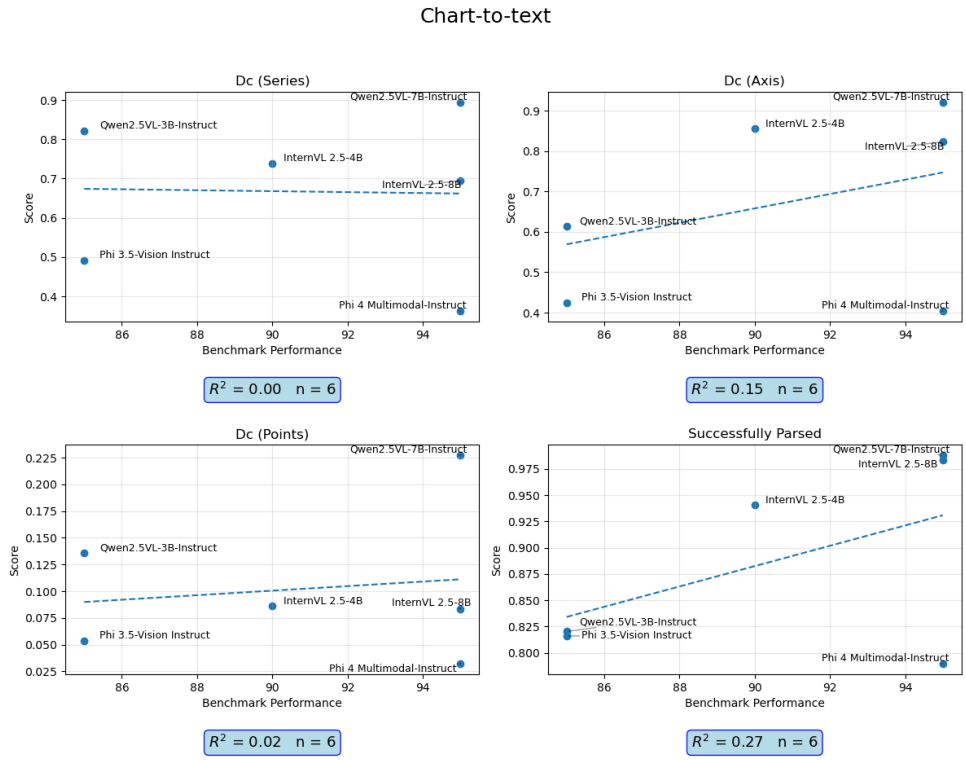


Figure 7: Correlation between Chart-to-Text benchmark scores and downstream structured extraction performance.

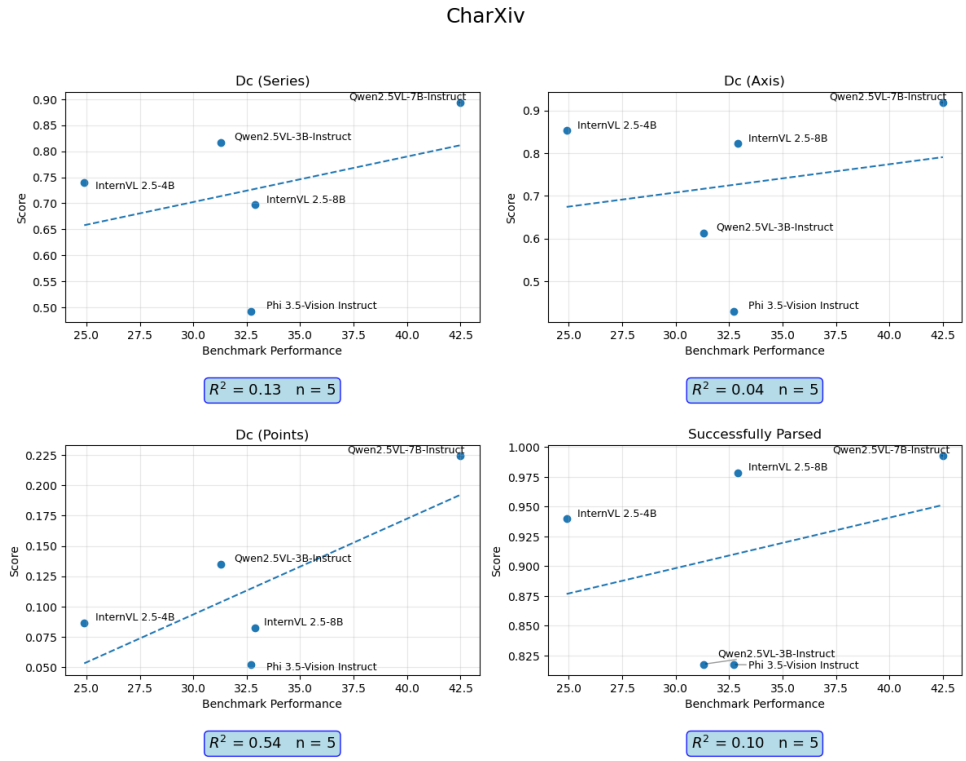


Figure 8: Correlation between CharXiv benchmark scores and downstream structured extraction performance.

Across benchmarks, correlation strength varies substantially. OCRBench v2 exhibits the strongest alignment with downstream performance, highlighting the importance of accurate textual localization and recognition for structured chart parsing. In contrast, Chart-to-Text and Chart-to-Table benchmarks show weaker and less consistent correlations, indicating that strong benchmark performance does not necessarily translate into robust scientific data extraction.

These results confirm that no single benchmark reliably predicts downstream structured extraction performance, reinforcing the necessity of task-specific evaluation protocols.

D Baseline Models Results Examples

To provide qualitative insight into model behaviour, this section presents representative output examples from baseline chart-extraction tools and general-purpose VLMs.

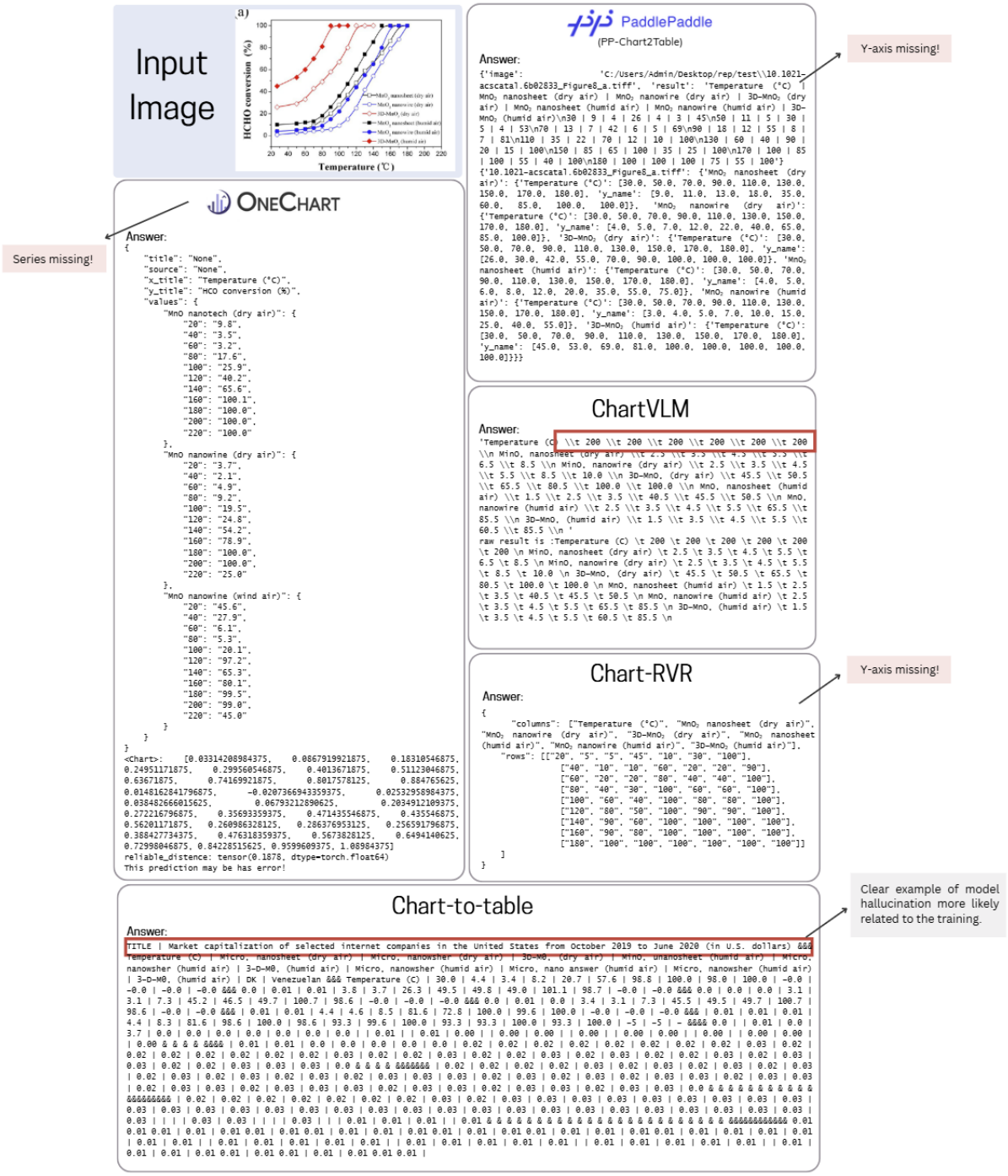


Figure 9: Representative output example from baseline chart-extraction tools.

Figure 9 shows typical failure cases observed in traditional chart-to-table systems, including:

- Missing y-axis labels
- Incorrect legend association

- Misaligned numerical outputs

- Non-parseable or incomplete CSV outputs

Inclusively, we tried to fine tune Unichart on our synthetic dataset but results were so inconsistent that it was impossible to parse, as it can be seen on the image below (Fig. 10 and 11).

{"Mn3O4": {"Temperature (°C)": [100.915327083514, 264.943397782381, 33.324645105028416, 264.941389127465, 356.45791178471666, 275.68952361860437, 280.5076322720908], "MEK conversion (%)": [4.882231949171084, 6.428789158551076, 7.873069161828522, 8.49847494100545, 10.73512861782814, 14.5067046414071752, 20.09051202626742б, 44.55527909832385, 97.9804876423582]}, "FeMnO": {"Temperature (°C)": [12.6980367640814, 264.943397782381, 33.32465105028416, 264.943387782381, 311.018541914676, 275.68952361860437, 280.5791178471666], "MEK conversion (%)": [6.082699923280562, 7.817273744798577, 6.55329376079886, 7.8178907762782, 7.269823095549837, 10.63794448182975, 26.260080272836]}, "FeO": {"Temperature (°C)": [100.915327083514, 141.1946536719466794, 141.11651165114309, 61.09489144613289, 47.1143093590310051З, 61.0935902763877, 61.09411546134764, 71.09243968050243963, 71, 2413224359712462143672172, 711], "MEK conversion (%)": [8006046998480935851569б5, 711], 141, -3312158115469836390224031546б8, ", -3312561546974031546б8, " | 39.940318463586390471546б8, wereGRAD35811547154715, 411], 411], "3033 spending on the 329.9482525641581154715811547З167, 4133254668, 411], 41.": " 406672165267154668154765266115476б5, 411], " | 35.809411547154715476527570415476б5, 411], " | 35.815471547154715476546316703154765, 411], " | 35.815471547654631673267326703154765, 47 " | 39.940871547654631670315476б5, 47 " | 47 | 39.940871547651547654631670315476б5, 47 " | 47 "408715476546316703154765, 47 "406703154765, 47 "406703154765, 47 "406703154765, 47 "4067036703670367, 47 "406703670367, "406703670367, "406703670367, "406703670367, "40670367, "406703670367, "406703670367, "406703670367, "406703670367, " "4067036703670367, " " "4067036703670367, " "4067036703670367, " " "4067036703670367, " " "40670367036703670367, " " " " "40670367036703670367, " "466807665, " " "4067036703670367036703670367, " " " " "466806807665, "466807665, " " "4067036703670367036703670367, " " " " "466807665, "466807665, "466807665, " "4067036703670367036703670367, " " " " "466807665, " "466804765, " "466804765, "4067036703670367036703670367, " " " "466807665, " " "466804765, "4668068, "4067036703670367036703670367, " " " " "466804765, "4668068, "4668068, "40670367036703670367036703670367, ", " " " " "4668068, "4668068, "40670367036703670367036703670367, ", " "466807664 ", "4665, "4668068, "4668068, "40670367036703670367036703670367036703670367, ", ", "4668076641367036703670367], ", ", "4697641367036703670367036703670367], ", ", ", "4697641367036703670367036703670367]}, ", ", ", "4697641367036703670367036703670367]}, ", ", "6413670367036703670367036703670367]}, ", ", "641367036703670367036703

Figure 10: Failure example of UniChart after fine-tuning on the synthetic dataset. The model produces structurally inconsistent and partially corrupted outputs, resulting in non-parseable predictions.

ANSWER:
{"PtNP/R-3DNG": {"Cycle number (n)": [2.6852143257479396, 11.49628087044336, 13.555716867825106, 14.962059058148582, 15.717639688248389, 17.583506976440736, 19.69036737182872, 20.676042216441043, 20.018730824405615, 34.3422217927115, 35.14741677719495б, 398.757474307433, 44.6622275873072, 45.053769879399069, 59.9153947702841, 64.4156567414791, 70.42439476419754, 79.82934788650902, 84.8292229732629, 89.69672759597106, 955.717423545534, 955.71722354423824, 955.744115225187, 9.940879548225187], "C37": {"Cycle number (n)": [2.6852143257479396, 11.49628087044336, 13.555716867825106, 15.717639688248389, 17.583506976440736, 19.962059058148582, 20.676042216441043, 34.342227587182872, 40.018730824405615, 45.05376987182872, 49.7319257479546479396, 494.0926574362434, | 431.9726157397642694719З6, | 431.97261572925436641193б, | 431.97261572925435735151936, | 47.972615729428, 47.9756157294329428, 47.972669471970383679, 47, 47.97296556556556556556556556556555, 47, 47, 47.76066947197064269471970642З6, 47, 47, 47.976256947976256426947197046556556556556555, 47, 47, 47, 479.97167064197064197064269471970642б947, 47, 47, 47 that of 6556556556556556556556556556556556556555, 47, 47, 47, 47, 47 that of | | | substitution (d)": [41973812294325754.765157423257294325432543254325432543254364 23, 47, 47, 47, 47.izedizedizedizedizedizedizedizedizedizedizedizedizedizedizedizedizedizedizedized, "CO)": "CO)": "CO)": "C37": "C37": "Cycle number (n1020190876, "Cycle number (n)": "Cycle number (n)": "Cycle number (n)": "Cycle number (n)": "Cycle number (n)": [2": [2": [2": [2": [2": [2": [2'α.1876.1876.1876.1576.1576.1576.1576.1576.1576, "Cy66351576, "Cy66351576, "Cycle number (N)": "Cycle number (N)": "Cycle number (N)": "Cycle number (N)": "Cycle number (N)": [2": [32576.0DWC811876.0 17.1576.0 17.1576.0 17.1576.0 17.1576.0 17.1576.0 17.107674, 17.107674, 17.107674, 17.107674157674, 17.107674, 17.107674, 17.107674, 17.107674.0 17.1076107674, 17.107674, 17.107674, 17.107674.0 17.107674, 17.107674.0 17., 17.10761076107674, 17., 17.107610761076107674, 17., 17.10761076107674, 17., 17.107610761076107674, 17., 17.107610761076107674, 47.0 17., 47.0 17., 47.0 17.1076107610761076107665157665157674, 47.0 17., 47.0 17., 47.0 17., 47.0 17.10761076107610761076653776653776653776653776, 47, 47.0 17., 47.0 17., 47.0 17., 47.0 17. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | Brazil and 31.2999232522222222222222222222222222222222222222222

Figure 11: Second example illustrating unstable generation behaviour of the fine-tuned UniChart model. Outputs show formatting breakdown and structural misalignment that hinder automated evaluation.

Figure 12 shows corresponding outputs after LoRA fine-tuning, illustrating improved point detection and structural consistency.

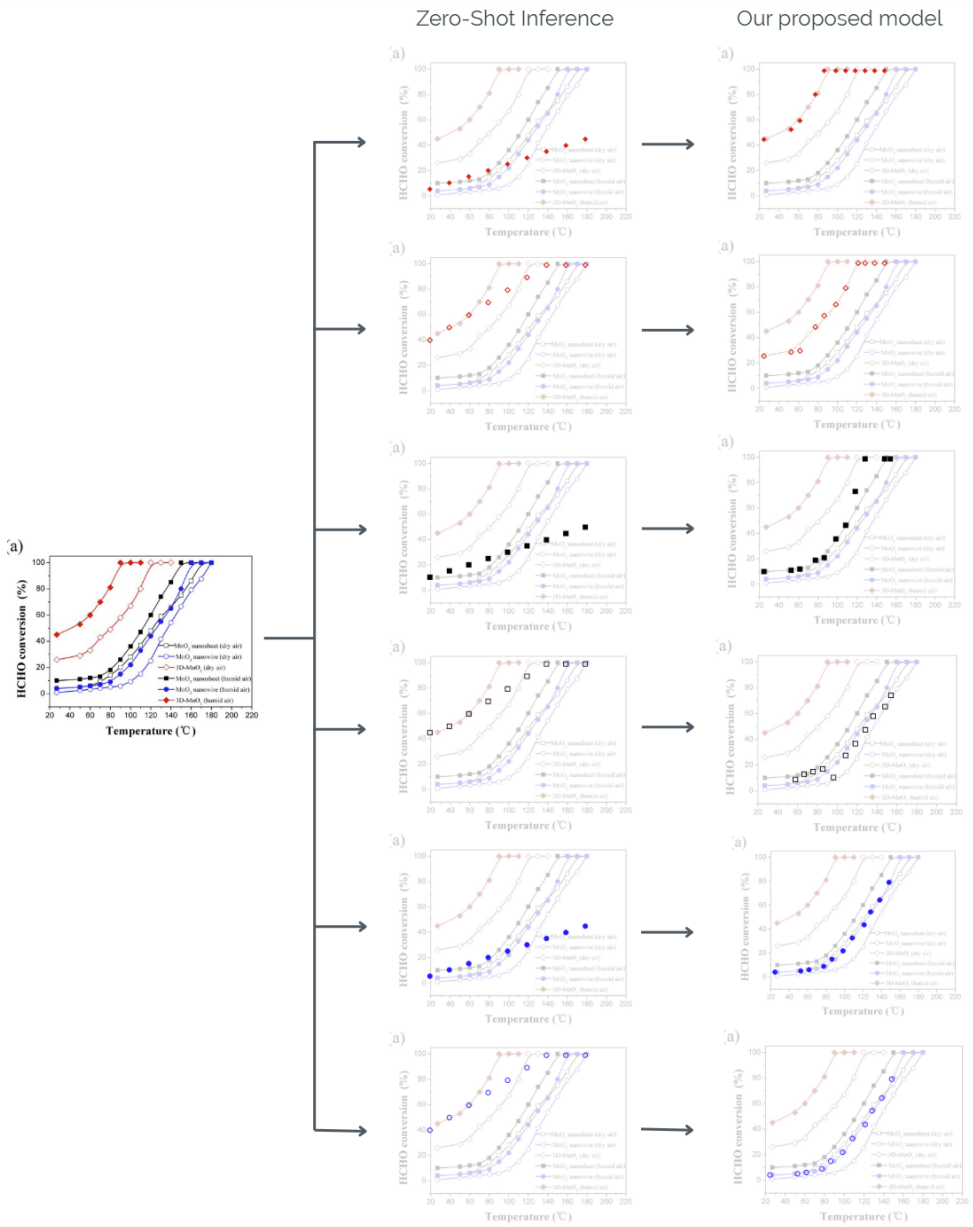


Figure 12: Proposed model vs zero shot inference.

Together, these qualitative examples complement the quantitative evaluation presented in the main manuscript and illustrate the practical advantages of the proposed fine-tuning strategy.

E Set-Up Details

The LoRA hyperparameters used during fine-tuning are summarized in Table 3, while the overall training configuration is reported in Table 4.

Table 3: LoRA parameters used for fine-tuning.

Parameter	Value
lora_alpha	32
lora_dropout	0.05
r	16
bias	none
target_modules	{gate_proj, up_proj, down_proj}
task_type	CAUSAL_LM

Table 4: Training parameters used for fine-tuning.

Parameter	Value
num_train_epochs	4
per_device_train_batch_size	4
per_device_eval_batch_size	4
gradient_accumulation_steps	8
gradient_checkpointing	True
optim	adamw_torch_fused
learning_rate	2e-4
lr_scheduler_type	constant
logging_steps	5
eval_steps	10
eval_strategy	steps
save_strategy	steps
save_steps	10
metric_for_best_model	eval_loss
greater_is_better	False
load_best_model_at_end	True
bf16	True
tf32	True
max_grad_norm	0.3
warmup_ratio	0.03
gradient_checkpointing_kwargs	{use_reentrant: False}
dataset_text_field	""
dataset_kwargs	{skip_prepare_dataset: True

All training and inference experiments were conducted on an NVIDIA A100 GPU partition.

The dataset contains a total of 1,400 images, split into 1,000 training samples, 200 validation samples, and 200 test samples used for evaluation.

The trained adapter weights are publicly available at `https://huggingface.co/datasets/LeaGabay/vlm4cat_data`.