Information Extraction from Diverse Charts In Materials Science

Anonymous authors
Paper under double-blind review

Abstract

The rapid advancements in machine learning necessitate parallel improvements in the size and quality of domain-specific datasets, especially in fields like materials science, where such datasets are often lacking due to the unstructured nature of real-world information. Despite the wealth of knowledge generated in this domain, much of it remains underutilized as experimental data is often buried in charts. In this paper, we curate two new benchmarks and introduce Relative Coordinate-Label Similarity (RCLS), a novel metric for measuring the state-of-the-art in extracting materials science data from scientific figures. We find that existing pretrained image-to-text Transformer based models for chart-to-table translation struggle with the diverse and complex nature of materials science figures, leading to issues such as inconsistent extraction of axis labels, irregular presentation of tabular data, and the omission of critical elements like legend labels from charts. We further fine-tune LLaMA 3.2-Vision 11B model to enhance its performance. Our study focuses on two subdomains of materials science, demonstrating both the successes and ongoing challenges in using multimodal models to extract scientific chart data.

1 Introduction

Extracting information from scientific charts and figures is a critical but challenging task for enabling large-scale information extraction in materials science (Schilling-Wilhelmi et al., 2025; Sayeed et al., 2023). Reasoning on visual-language tasks, such as chart-to-table conversion, has gained significant attention in recent literature due to its potential for structuring complex visual data (Huang et al., 2024; Islam et al., 2024). Zaki et al. (2022) demonstrated the utility of WebPlotDigitizer V4 (Automeris, 2024), an open-source tool that converts data from plots into numerical values (Zaki et al., 2022). However, this tool requires the manual processing of images, limiting its scalability and efficiency.

LLMs have shown promise in visual reasoning tasks (Achiam et al., 2023; Seitl et al., 2024; Xu et al., 2024), but they often struggle with understanding charts (Mukhopadhyay et al., 2024). Their effectiveness in extracting structured information from domain-specific scientific charts remains an underexplored area. This work seeks to bridge this gap by focusing on the extraction of information from materials science charts.

Several recent works have explored information extraction from scientific charts and figures. DePlot (Liu et al., 2022) transforms the image of a plot or chart to a linearized table, but has limitations in translating to out-of-distribution data (Nowak et al., 2024). MatViX (Khalighinejad et al., 2024) proposes a multimodal information extraction framework for visually rich scientific articles, demonstrating the potential for extracting key insights from figures in the domain of materials science. However, MatViX does not directly evaluate the task of chart-to-table conversion, limiting its utility for complete numerical data extraction. Taniguchi & Lindsey (2025) highlight limitations when it comes to extracting detailed spectral information from charts (Taniguchi & Lindsey, 2025). Similarly, ChartX and ChartVLM (Xia et al., 2024) introduce chart reasoning models and benchmarks, though their datasets predominantly feature generic, domain-agnostic charts, lacking the complexity of materials science figures.

Existing methods largely overlook the diversity and complexity of scientific figures. Using the exemplar of the domain of materials science, visuals often represent multi-dimensional experimental conditions, material properties, or processing parameters. To address this, we evaluate information extraction models on a curated dataset of materials science charts and introduce strategies to improve performance in this domain. The resulting tabular data can then be used to prompt LLMs for data restructuring (Circi et al., 2024) or question answering as shown in DePlot.

In this work, we aim to advance information extraction from materials science figures by:

1. Introducing two diverse datasets of real materials science charts, **PolyCompChartIE** and **MetalThermoChartIE**, for evaluating chart-to-table conversion models, addressing the current lack of domain-specific benchmarks (Dredze et al., 2024).
2. Proposing a new evaluation metric, **Relative Coordinate-Label Similarity (RCLS)**, designed specifically to assess numerical data extraction accuracy from scatterplot charts.
3. Providing insights into the limitations of existing visual-language models in handling complex materials science figures and outlining future directions for performance improvement.

2 Related Work

2.1 Datasets and Tasks

Several chart-focused datasets have been proposed to support visual-language reasoning, including PlotQA (Methani et al., 2020), ChartQA (Masry et al., 2022), and FigureQA (Kahou et al., 2017). FigureQA consists of over 1 million procedurally generated synthetic figures alongside yes/no questions pertaining to chart features and data. However, this data lacks real-world visual complexity and has labels that are only useful for QA tasks, failing to extend to the chart-to-table task. PlotQA improved upon FigureQA by incorporating real plot images extracted from scientific documents from sources such as arXiv, as well as including the underlying chart metadata which permits use for the chart-to-table task. However, these figures remain largely generic and do not incorporate the domain-specific complexity associated with materials science figures. ChartQA provides a large set of synthetic and real-world chart question-answering data, for use in learning and evaluating model performance on a greater variety of visual-language tasks for bar, line, and pie charts, such as chart summarization, chart-to-code generation, and open-ended question answering. While this dataset reflects a greater diversity of real-world chart examples and can be evaluated on a more comprehensive set of tasks, a representative real-world dataset that tests existing and future models' capabilities in extracting tabular data from complex materials science figures, particularly scatter-plots, is still missing from the literature.

2.2 Benchmarks

Recent efforts have sought to standardize evaluation across visual reasoning tasks. MaCBench (Alampara et al., 2024) introduces a multimodal benchmark covering scientific text, laboratory scenes, and microscopy, but includes limited chart-based reasoning. Wu et al. (2024) introduced VISCO, a benchmark testing self-improvement in visual reasoning, further revealing the limitations of current visual-language models on complex data visualizations. Islam et al. (2024) provide an extensive evaluation of large vision-language models(LVLMs) across many chart reasoning tasks, including chart-to-table extraction, using metrics such as Relative Number Set Similarity (RNSS) and Relative Mapping Similarity (RMS) discussed later. While these benchmarks effectively evaluate generic chart datasets on a number of tasks, they do not address the unique challenges posed in chart-to-table conversion for our more complex materials science figures. To bridge this gap, we propose a new evaluation metric, Relative Coordinate-Label Similarity (RCLS), which measures the accuracy of coordinate-value and label matching in reconstructed tables, providing a more rigorous assessment of chart-to-table extraction performance.

2.3 Models

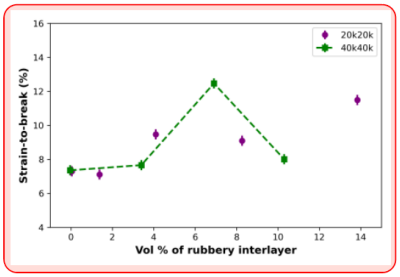
Early approaches to chart understanding and visual reasoning tasks used static visual-language pipelines, but recent work has been largely dominated by end-to-end transformer-architecture models. DePlot (Liu et al., 2022) uses a visual encoder and text decoder to map plots directly to tables. However, its performance degrades greatly on out-of-distribution data (Nowak et al., 2024). UniChart and ChartLLaMA (Xia et al., 2024) leverage fine-tuned LLMs to perform visual reasoning tasks on charts, but rely heavily on datasets with simple charts and clear numerical overlays, greatly reducing the difficulty of data point extraction. Large Vision-Language Models (LVLMs), such as GPT-4V (Achiam et al., 2023), Claude 3.5 and Gemini have demonstrated strong capabilities in chart reasoning tasks (Anthropic, 2024; DeepMind, 2024; Islam et al., 2024), but still struggle with precise numerical extraction and semantic hallucination, particularly when figures do not contain explicitly labeled data points. These limitations are exacerbated in materials science figures, where layout, symbol density, and overlapping data points demand more specialized training and evaluation procedures, demonstrating the need for domain-specific model development.

3 MatSci Chart Information Extraction Benchmark

In this section, we describe our problem, dataset preparation, and evaluation metrics.

3.1 Problem Definition

In this work, we focus on the extraction of tabular data from materials science charts, specifically those related to polymer composites and thermophysical property data of metal systems. These charts typically represent a material property value, such as electrical conductivity or tensile strength, on the y-axis, and a variable that affects this property, such as temperature, pressure, or compositional data, on the x-axis. Each scatter plot's set of data points has varying density, clarity, and size. Additionally, these plots frequently include categorical labels that provide further contextual information. These labels are indicated through various visual elements, such as different colors, shapes of point marks, or string labels, and may represent distinct experimental conditions, experiment numbers, additional factors affecting the property, or citations of data points sourced from other studies. Our goal is to extract the x and y axis labels, the numerical coordinate values for each data point on the x and y axes, and any corresponding labels. An example can be seen in Figure 3.



Vol% of Rub.	Strain-to-break	
0.045	7.297	20k20k
1.375	7.117	20k20k
4.096	9.473	20k20k
8.264	9.096	20k20k
13.84	11.495	20k20k
-0.043	7.357	40k40k
3.399	7.663	40k40k
6.908	12.471	40k40k
10.3	8.013	40k40k

Figure 1: Illustration of the chart-to-table extraction task. Left: A masterials science scatter plot showing strain-to-break (%) versus volume percentage of rubbery interlayer for two sample categories (20k20k and 40k40k). Right: The extracted data in tabular format with column headers and values organized by x-axis values, y-axis values, and categorical labels.

3.2 Dataset Generation

We introduce two datasets from distinct domains within materials science: polymer composites and thermophysical property data of metal systems. These are annotated by humans using WebPlotDigitizer manually. The average number of data points per chart is 17.11, with the minimum number of data points being 3 and the maximum number of data points being

73. For charts that include categorical labels—providing further contextual information for specific data points—the average number of unique labels is 3.40. The dataset includes 52 charts with categorical labels and 24 charts without labels.

Polymer composite article charts. (PolyCompChartIE) This dataset consists of 77 charts extracted from research articles in the domain of polymer composites. These charts are used exclusively for testing purposes, providing a benchmark to evaluate the effectiveness of our methods in this specific context.

Thermophysical property data of metal systems article charts. (MetalThermoChartIE) This dataset consists of 620 charts sourced from articles focused on thermophysical properties of metal systems. We divide this dataset into training, validation, and testing subsets with a split of 60%, 20%, and 20%, respectively. In the test set, the average number of data points per chart is 31.17, with the minimum number of points being 3 and the maximum number of points being 134. For charts that include categorical labels, the average number of unique labels is 3.53. The dataset includes 58 charts with categorical labels and 66 charts without labels.

Plot recreation using GPT-4o for code generation.

The articles in our dataset span publications from 1920 to 2018, resulting in a wide range of image qualities. Older plots, often hand-drawn, exhibit significant noise, which poses challenges for automated data extraction. To address this, we sought to recreate the original charts programmatically, generating high-quality, standardized visualizations.

We employed GPT-4o to reconstruct the provided plots using Python and Matplotlib. Prior research (Zhang et al., 2024) indicates that vision-language models (VLMs) struggle to infer exact numerical values without access to raw data in a human evaluation. To mitigate this, we included tabular annotations of the raw data in the input, ensuring a more faithful reproduction of the original plots. The prompt, which incorporates a chain-of-thought approach, is provided in section A.1. The model-generated code was then executed to produce the visualizations.

To further refine the results for the test set, we implemented an iterative improvement process. We supplied the model with the original chart, the initial generated plot, and the corresponding code, prompting it to enhance the accuracy and fidelity of the reproduction. The prompt can be found in section A.1. An example illustrating the original plot, the initially generated plot, and the improved version can be found in subsection A.2. Each version was manually reviewed, and the most representative plot was selected. In cases where the complexity of the figure—such as highly irregular phase diagrams—exceeded the model's capability, the generated plots were discarded.

For the PolyCompChartIE dataset, 68 out of 77 plots were successfully reconstructed, while for the MetalThermoChartIE dataset, 93 out of 124 plots were accurately regenerated. This reconstructed dataset enables us to assess information extraction performance on both idealized, noise-free plots and original, often degraded scientific charts. The recreated charts will be made publicly available, providing a valuable resource for the community. However, the original charts remain subject to copyright restrictions and cannot be widely shared.

3.3 Evaluation

Some prior research has been conducted in constructing metrics for evaluating table similarity. Namely, Liu et al. (2022) have proposed *Relative Mapping Similarity* (`RMS`). `RMS` makes key improvements in incorporating positional relationships between numerical data points, accepting more flexibility in table representation, and reflecting losses in both precision and recall over a previous metric proposed in (Luo et al., 2021) and introduced in (Masry et al., 2022) named *Relative Number Set Similarity* (`RNSS`). However, because `RMS` was primarily used with bar and pie charts, it treats tables as an unordered set of mappings from column and row headers to single values, a structure that does not transfer sensibly to scatter plots. We formally introduce *Row-Wise Mapping Similarity* (`RWMS`), a metric found in the google-research GitHub repository associated with (Liu et al., 2022) but not explicitly described in the corresponding publication. While `RWMS` is well-suited for comparing tables derived from scatter

plots, its current implementation relies on a relative distance calculation that introduces bias based on chart axis ranges and inconsistently credits errors at different relative distances (See Figure 3). To address these issues, we propose a new metric, *Relative Coordinate-Label Similarity* (RCLS), which we formally define in the following section alongside a detailed comparison with RWMS.

Row-Wise Mapping Similarity (RWMS). With this metric, a table is viewed as an unordered collection of row objects, written as $\mathbf{p}_i = (p_i^1, p_i^2, \ldots, p_i^n)$ and $\mathbf{t}_i = (t_i^1, t_i^2, \ldots, t_i^m)$ for each row in the predicted table $\mathbf{P} = \{\mathbf{p}_i\}_{1 \le i \le N}$ and the target table $\mathbf{T} = \{\mathbf{t}_j\}_{1 \le j \le M}$, respectively. To calculate the distance D_θ between numerical values, the entries' difference is normalized by the target value: $D_\theta(\mathbf{p}, \mathbf{t}) = \min(1, \|\mathbf{p} - \mathbf{t}\| / \|\mathbf{t}\|)$. Distances above a certain threshold θ are set to 1, the maximum value. Textual entries are compared using *Normalized Levenshtein Distance* (NL_ϕ), where distance scores above ϕ are set to 1, the maximum value, restricting matching to text pairs with sufficient similarity. If the number of elements in $\mathbf{p}$, n, is not equal to the number of elements in $\mathbf{t}$, m, a row-similarity score of 0 is returned. Otherwise, the row similarity score is calculated as the product of each element-wise comparison,

$$S_{\phi,\theta}(\mathbf{p}, \mathbf{t}) = \prod_{k=1}^{K} \prod_{l=1}^{L} \left(1 - \mathsf{NL}_\phi(p_k, t_k)\right) \left(1 - D_\theta(p_l, t_l)\right),$$

for a $(\mathbf{p}, \mathbf{t})$ pair with K textual entries and L numerical entries. Using the cost function $(1 - S_{\phi,\theta})$ across the set of all possible $(\mathbf{p}, \mathbf{t})$ combinations produces an $M \times N$ similarity matrix, with which we can identify the minimal cost matching $\mathbf{X} \in \mathbb{R}^{M \times N}$ between the row objects (in the form of a binary matrix). The precision and recall are then calculated as:

$$\mathsf{RWMS}_{\text{precision}} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{M} \mathbf{X}_{ij} S_{\phi,\theta}(p_i, t_j)}{N}, \tag{1}$$

$$\mathsf{RWMS}_{\text{recall}} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{M} \mathbf{X}_{ij} S_{\phi,\theta}(p_i, t_j)}{M}. \tag{2}$$

The F_1 score is recorded as the harmonic mean of the precision and recall. This metric also "considers the table and its transposition and returns the one with a higher RMSF1 score".

The RWMS metric has a couple of important limitations. Using a wide range of values for $\mathbf{p}$ and $\mathbf{t}$ in the calculation of D_θ quickly reveals that this method is heavily biased in favor of larger values relative to the range of the chart. For example, on a chart whose values range from 0 to 100, a prediction of 2 against a target of 1 would result in a relative distance of 1, whereas a prediction of 99 against a target of 100, equivalent in error magnitude, receives a relative distance of 0.01. While both of these predictions are relatively accurate given the wide range of the chart, only one is considered a match. This bias skews evaluation greatly based on distribution of points on the chart, and this becomes a substantial issue when evaluating charts of diverse x and y axis ranges.

Furthermore, for our specific chart type of interest, scatter plots, comparing numerical values on an element basis does not treat errors of the same two-dimensional magnitude consistently. A prediction-target row pair that is within the relative error threshold θ in both x and y directions individually may or may not be within that relative error threshold when treated as a pair of (x, y) coordinates, a more intuitive and appropriate data structure for scatterplot points. We argue that our metric RCLS addresses these concerns and is more suitable for scatter plot data table comparison.

Relative Coordinate-Label Similarity (RCLS). To address these limitations, we propose RCLS, which also views tables as unordered collections of row objects. However, the structure of our scatter plot data is largely consistent. In order to reflect x and y values appropriately corresponding to their axis labels as well as the legend label corresponding to each point, we adopted the convention of using the first column for the x-axis label and x values, the second column for the y-axis label and y values, and the third column for the legend labels, if they exist, for individual points. This convention addresses the frequent use of legend labels in scatter plots, which essentially create a third degree of freedom. It can be tempting to use this label as an additional column header, to avoid repeating it for many coordinates. **However,**

this style of annotation prevents us from correctly aligning the x- and y-axis labels, and thus using the label on a row basis is ideal. With this convention, it is useful notation to denote these row entries as $\mathbf{p}_i = (\mathbf{p}_i[x, y], l_i)$, where $\mathbf{p}_i[x, y]$ indicates a coordinate pair and l_i represents its corresponding textual label, left as '' if none exists in the particular chart. In addition, a "headers" row entry is added, written as $p_h = (p_h^x, p_h^y)$ and $t_h = (t_h^x, t_h^y)$ for prediction headers and target headers respectively, containing two elements, the x-axis label and y-axis label. Calculation of distance between textual entries remains the same as with RWMS, using the *Normalized Levenshtein Distance* ($\mathtt{NL}_\phi$). The distance between $\mathbf{p}_i(x, y)$ and $\mathbf{t}_i(x, y)$ is calculated using the 2-dimensional Euclidean distance formula, with the x and y contributions being normalized by $x_{\text{range}} = x_{\max} - x_{\min}$ and $y_{\text{range}} = y_{\max} - y_{\min}$ respectively:

$$D_\theta(\mathbf{p}, \mathbf{t}) = \sqrt{\left(\frac{t_x - p_x}{x_{\text{range}}}\right)^2 + \left(\frac{t_y - p_y}{y_{\text{range}}}\right)^2}.$$

Using the Euclidean distance formula ensures that the threshold θ for coordinate similarity is applied radially, rather than first in the x direction, and then in the y direction, which creates a rectangular acceptable error bound. Additionally, normalizing by the x and y ranges instead of the singular target value improves consistency in comparison of data points in different locations within a single chart as well as for diverse chart axis ranges. Illustration of the comparison of error bounds between RWMS and RCLS can be seen in subsection A.4.

We compute the similarity score between two row objects as $S_{\phi,\theta} = (1 - \mathtt{NL}_\phi)(1 - D_\theta)$, such that when both labels and coordinates are similar, $S_{\phi,\theta}$ is close to 1. The prediction and target "headers" are also compared, and given a similarity score $S_{\phi,\theta} = (1 - \mathtt{NL}_\phi(p_h^x, t_h^x))(1 - \mathtt{NL}_\phi(p_h^y, t_h^y))$. We then construct the minimal cost matching $\mathbf{X} \in \mathbb{R}^{M \times N}$ as in RWMS, and calculate the $\mathtt{RCLS}_{\text{precision}}$ and $\mathtt{RCLS}_{\text{recall}}$ using equations (1), (2). As before, the $\mathtt{RCLS}_{F1}$ score is the harmonic mean of the precision and recall. We found that comparing transpositions with our style of annotation did not improve any of these metrics through genuine table similarity, thus we do not compare table transpositions when computing RCLS.

Number-Only Evaluation. The problem of information extraction from materials science scatter plots has two primary extraction goals: To accurately extract the numerical data-points, and to extract the textual elements, namely the axis labels and legend labels. While the previously presented metric RCLS incorporates both of these in its evaluation, it does not provide a clear indication of the specific performance of either of these two tasks individually. Cliche et al. (2017) introduced a metric which looks only at numerical datapoint extraction, viewing tables as an unordered list of coordinates written as $\mathbf{p}_i = (p_i^x, p_i^y)$ and $t_j = (t_j^x, t_j^y)$ for each row in the predicted table $\mathbf{P} = \{\mathbf{p}_i\}_{1 \le i \le N}$ and the target table $\mathbf{T} = \{\mathbf{t}_j\}_{1 \le j \le M}$, respectively. The pair of distances $D_x = \frac{|X_{\text{pred}} - X_{\text{true}}|}{\Delta X_{\text{true}}}$ and $D_y = \frac{|Y_{\text{pred}} - Y_{\text{true}}|}{\Delta Y_{\text{true}}}$ are computed for all possible prediction-target pairs, and pairings with $D_x \le 0.02$ and $D_y \le 0.02$ are considered true positives. Firstly, the target points' nearest neighbor predictions are selected and true positives are determined. If a target prediction pair satisfies the true positive conditions, both are removed from the set to ensure each prediction is counted at most once. This process is repeated for remaining points until none satisfy the true positive conditions, or there are no more points. Finally, the precision, recall, and F_1 are computed as TP/N, TP/M, and their harmonic mean, respectively.

To better understand sources of error and model performance, we introduce a modification of RCLS called *Relative Coordinate Similarity* (RCS) which also views tables as unordered lists of coordinates, but utilizes the distance formula described in RCLS. Header rows are not compared with RCS, and the similarity score is defined as $S_{\phi,\theta} = 1 - D_\theta$, where D_θ is calculated using the same distance formula as in RCLS. Once again, the minimal cost matching is performed and the $\mathtt{RCS}_{\text{precision}}$, $\mathtt{RCS}_{\text{recall}}$, $\mathtt{RCLS}_{\text{F1}}$ are calculated with equations (1), (2), and their harmonic mean, respectively.

4 Experiment

We test GPT-4o, o1, LLaMA 3.2-Vision 11B and 90B, and Qwen 2.5 VL models. The prompt used to extract the tabular data from charts can be found in section A.1. We find that both the Llama and Qwen models are unable to reliably produce CSVs. We parse their outputs using Gemini 2.0 Flash-Lite. We create a training set based on 75% of our original chart images.

We used OpenAI API to prompt the o1 and GPT-4o models. For GPT-4o model, we set the maximum token size to 4096.

	PolyCompChartIE						**MetalThermoChartIE**					
	Original Chart (76)			Recreated Chart (68)			Original Chart (124)			Recreated Chart (93)		
Model	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
o1	**45.08**	**48.81**	**46.30**	-	-	-	**23.37**	**20.98**	**20.59**	-	-	-
GPT-4o	41.01	35.46	36.36	43.40	38.89	39.77	**28.76**	15.50	18.14	42.27	22.60	25.67
(Models using Gemini for CSV extraction):												
LLaMa 3.1 Vision 11B + Finetune	40.83	41.08	39.72	-	-	-	**66.55**	42.63	**48.11**	-	-	-
LLaMa 3.1 Vision 11B	42.02	44.99	42.50	-	-	-	43.79	45.12	42.71	-	-	-
Qwen2.5-VL	**44.62**	**48.02**	**45.01**	-	-	-	46.02	**46.94**	45.00	-	-	-

Table 1: Precision (P), Recall (R), and F_1-score (F_1) for different models on the PolyCompChartIE and MetalThermoChartIE datasets, comparing original and recreated charts. "Models using Gemini for CSV Extraction" means that we used Gemini 2.0 Flash-Lite to parse outputs from Llama and Qwen into CSVs, since they did not reliably produce outputs in the correct format.
For PolyCompChartIE, we evaluate models on all 76 test images. Please see table Table 2 for a numbers-only comparison, which improves the results significantly and shows that Qwen is primarily out-performing Llama on text extraction, rather than information extraction from the plotted points themselves.

To understand the difference between the original charts and the recreated ones, we used the GPT-4o model. We observe that the recreated charts scored higher in both of the datasets. With the MetalThermoChartIE dataset specifically, we observe significant improvement from 18.14 F1 score to 25.67, likely due to this dataset's increased diversity and generally higher average number of data points. The o1 model performed the best on the PolyCompChartIE set. When we finetune the LLaMa 3.1 Vision 11B model on the MetalThermoChartIE training set, we obtain the highest F1 score of 48.11. Notably, the Qwen2.5-VL model's performance is very close to that of the best models, even the finetuned LlaMa Model, in both datasets.

5 Discussion

5.1 Error analysis

Two examples from the MetalThermoChartIE dataset highlighting the challenges faced by GPT-4o model can be found in Figure 2. A primary source of error in the use of LVLMs for the chart-to-table task is numerical data point extraction. While LVLMs show strong capabilities in extracting the textual components of our materials science figures, namely the axis labels and legend labels, they frequently hallucinate and omit data points, generally predicting them at convenient intervals along the x-axis, such as the tick marks, and providing y-coordinates that reflect the general trend of the data, i.e. whether it is increasing or decreasing, often in a linear fashion, but not fine extraction. Additionally, data points that are clustered tightly or overlapping are generally not recorded, leading to incomplete extracted data tables.

We observe substantial performance improvements after finetuning Llama 3.1 Vision 11B, even on a limited number of domain-specific training samples. This suggests a lack of

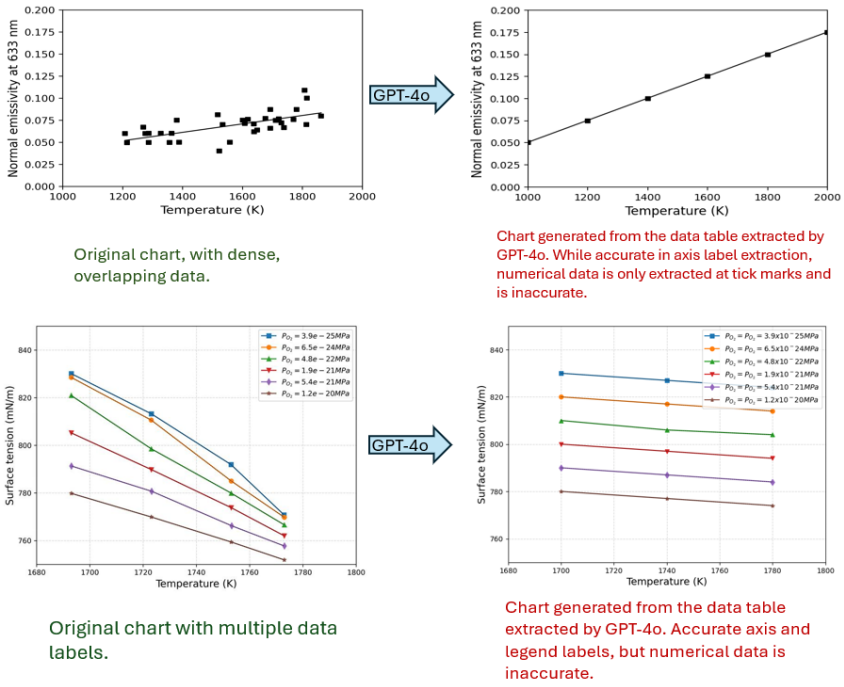


Figure 2: Examples showing GPT-4o data extraction challenges on the MetalThermoChartIE dataset. Top: While GPT-4o correctly identifies axis labels, it incorrectly linearizes the scattered data points. Bottom: GPT-4o accurately reproduces axis labels and legend information but introduces numerical inaccuracies in the extracted data values.

complex scientific charts in Llama's pre-training and finetuning data. This may be expected as materials science chart data is often subject to copyrights, can be difficult to acquire, and currently requires intensive manual annotation. This highlights a significant opportunity for synthetic data generation to bridge this gap and enhance model capabilities for this domain.

5.2 Original chart vs recreated code generated chart

Through human evaluation, we found that GPT-4o performs well in generating code to accurately recreate charts when the underlying data is provided. However, in some cases the generated plots contained errors that altered the visual representation of the data—potentially compromising accurate information extraction. As a result, 8 charts were removed from the PolyCompChartIE dataset, and 31 charts were excluded from the PolyCompChartIE dataset.

6 Conclusion

Extracting data locked in complex materials science charts remains challenging, even for frontier models like GPT-4o and o1. To address this, we introduced two domain-specific benchmarks and the **RCLS** metric tailored for scatter plot evaluation. Our evaluation revealed that current vision language models struggle with the noise and diversity of real-world scientific charts, exhibiting numerical and labeling inaccuracies, although performance improves markedly on cleaner, recreated versions. We observe that fine-tuning Llama 3.1 Vision 11B brought large performance gains even with limited data. Our work provides essential insights and resources that support the development of more accurate extraction methods—ultimately contributing to accelerated materials discovery through improved automation.

Ethics Statement

We do not believe there are any ethical issues associated with this research.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Nawaf Alampara, Mara Schilling-Wilhelmi, Martiño Ríos-García, Indrajeet Mandal, Pranav Khetarpal, Hargun Singh Grover, NM Krishnan, and Kevin Maik Jablonka. Probing the limitations of multimodal language models for chemistry and materials research. *arXiv preprint arXiv:2411.16955*, 2024.

Anthropic. Claude 3.5. https://www.anthropic.com/index/claude-3-5, 2024. Accessed: March 2025.

Automeris. Webplotdigitizer: A web-based tool to extract data from plots, images, and maps, 2024. URL https://github.com/automeris-io/WebPlotDigitizer.

Defne Circi, Ghazal Khalighinejad, Anlan Chen, Bhuwan Dhingra, and L Catherine Brinson. How well do large language models understand tables in materials science? *Integrating Materials and Manufacturing Innovation*, 13(3):669–687, 2024.

Mathieu Cliche, David Rosenberg, Dhruv Madeka, and Connie Yee. Scatteract: Automated extraction of data from scatter plots. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18–22, 2017, Proceedings, Part I 10*, pp. 135–150. Springer, 2017.

Google DeepMind. Gemini 1.5. https://deepmind.google/technologies/gemini/, 2024. Accessed: March 2025.

Mark Dredze, Genta Indra Winata, Prabhanjan Kambadur, Shijie Wu, Ozan İrsoy, Steven Lu, Vadim Dabravolski, David Rosenberg, and Sebastian Gehrmann. Academics can contribute to domain-specialized language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5100–5110, 2024.

Kung-Hsiang Huang, Hou Pong Chan, Yi R Fung, Haoyi Qiu, Mingyang Zhou, Shafiq Joty, Shih-Fu Chang, and Heng Ji. From pixels to insights: A survey on automatic chart understanding in the era of large foundation models. *arXiv preprint arXiv:2403.12027*, 2024.

Mohammed Saidul Islam, Raian Rahman, Ahmed Masry, Md Tahmid Rahman Laskar, Mir Tafseer Nayeem, and Enamul Hoque. Are large vision language models up to the challenge of chart comprehension and reasoning? an extensive investigation into the capabilities and limitations of lvlms. *arXiv preprint arXiv:2406.00257*, 2024.

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

Ghazal Khalighinejad, Sharon Scott, Ollie Liu, Kelly L Anderson, Rickard Stureborg, Aman Tyagi, and Bhuwan Dhingra. Matvix: Multimodal information extraction from visually rich articles. *arXiv preprint arXiv:2410.20494*, 2024.

Fangyu Liu, Julian Martin Eisenschlos, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Wenhu Chen, Nigel Collier, and Yasemin Altun. Deplot: One-shot visual language reasoning by plot-to-table translation. *arXiv preprint arXiv:2212.10505*, 2022.

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*, pp. 1917–1925, 2021.

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

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

Srija Mukhopadhyay, Adnan Qidwai, Aparna Garimella, Pritika Ramu, Vivek Gupta, and Dan Roth. Unraveling the truth: Do vlms really understand charts? a deep dive into consistency and robustness. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 16696–16717, 2024.

Averi Nowak, Francesco Piccinno, and Yasemin Altun. Multimodal chart retrieval: A comparison of text, table and image based approaches. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5488–5505, 2024.

Hasan M Sayeed, Wade Smallwood, Sterling G Baird, and Taylor D Sparks. Quantifying the distribution of materials data types in scientific literature across text, tables, and figures. 2023.

Mara Schilling-Wilhelmi, Martiño Ríos-García, Sherjeel Shabih, María Victoria Gil, Santiago Miret, Christoph T Koch, José A Márquez, and Kevin Maik Jablonka. From text to insight: large language models for chemical data extraction. *Chemical Society Reviews*, 2025.

Filip Seitl, Tomáš Kovářík, Soheyla Mirshahi, Jan Kryštfek, Rastislav Dujava, Matúš Ondreička, Herbert Ullrich, and Petr Gronat. Assessing the quality of information extraction. *arXiv preprint arXiv:2404.04068*, 2024.

Masahiko Taniguchi and Jonathan S Lindsey. Acquisition of absorption and fluorescence spectral data using chatbots. *Digital Discovery*, 4(1):21–34, 2025.

Xueqing Wu, Yuheng Ding, Bingxuan Li, Pan Lu, Da Yin, Kai-Wei Chang, and Nanyun Peng. Visco: Benchmarking fine-grained critique and correction towards self-improvement in visual reasoning. *arXiv preprint arXiv:2412.02172*, 2024.

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

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen. Large language models for generative information extraction: A survey. *Frontiers of Computer Science*, 18(6):186357, 2024.

Mohd Zaki, NM Anoop Krishnan, et al. Extracting processing and testing parameters from materials science literature for improved property prediction of glasses. *Chemical Engineering and Processing-Process Intensification*, 180:108607, 2022.

Zhehao Zhang, Weicheng Ma, and Soroush Vosoughi. Is gpt-4v (ision) all you need for automating academic data visualization? exploring vision-language models' capability in reproducing academic charts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 8271–8288, 2024.

A Appendix

A.1 Prompts

Prompt for Chart Recreation Below, we provide the prompt used to initially re-create charts.

Chart Recreation Prompt

This is a data visualization figure from an academic paper. Please write Python code to draw the exact same plot as this one and save it as a PNG file with 300dpi.

Here are the specific x and y values for the datapoints included in the figure with their corresponding labels, if it exists:

{datapoints}

Let's think step-by-step.

Prompt for Chart Refinement Below is the prompt used to refine re-created charts.

Recreated Chart Refining Prompt

I have written a Python script to generate a plot that matches the original plot I provided. However, the output plot does not look identical to the original. Please review the differences between the two plots and modify the code to ensure the generated plot matches the original in every detail. Pay attention to aspects like marker size, color, line thickness, grid style, axis labels, data labels and their position, positions of the ticks on the axes, figure dimensions, and any other visual elements.

Original code:
{code}

Please provide the updated code to make the generated plot match the original plot exactly.

Chart information extraction prompt The prompt below was used to extract data from charts.

Chart Information Extraction Prompt

Extract the numerical data from the figures in image format.
The aim is to convert the plots into CSV tables. The input plots are coming from materials
science articles and may include line graphs, scatter plots, or bar charts.

For each input image:
1. Identify the x and y axis labels.
2. Extract data points from the plot. The data points do not need to align with the ticks
in the axis. I want to include all the data points that appear in the plot.
3. If there are multiple data series in a single plot, identify and label them.

The output should be a CSV file, formatted as follows:
- First row: x-axis label, y-axis label
- Subsequent rows: x value, y value, series label (if applicable)

If there's no series label for a data point, the output should have only 2 columns.

Example output 1:
Graphene nanoplatelet (wt%),Flexural modulus (GPa),Series
0,3.1,GnP-C750/epoxy
3,3.2,GnP-C750/epoxy
5,3.25,GnP-C750/epoxy
0.5,3.1,GnP-5/epoxy
3.7,3.6,GnP-5/epoxy
4.6,3.9,GnP-5/epoxy

Example output 2:
SiO2 content (wt0,375
1,408
2,435
5,480
10,435
15,425

A.2 Example of recreating the original plot using code generated by GPT-4o.

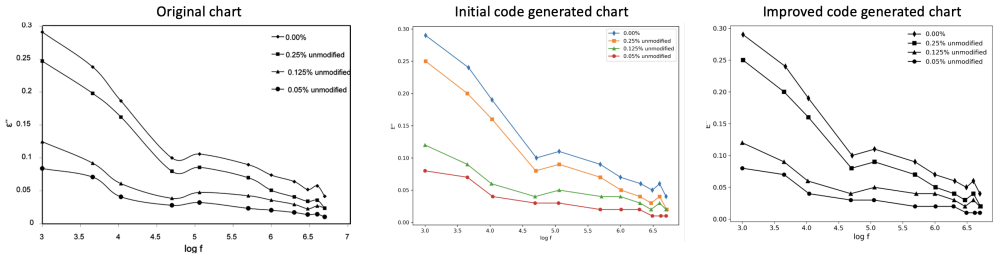


Figure 3: Illustration of chart recreation using code. Left: Original materials science chart from the MetalThermoChartIE dataset. Middle: Initial chart generated after providing GPT-4o with the underlying numerical data. Note the incorrect colors and extraneous rectangle around the legend. Right: Improved chart after feeding back the original plot and initial generated chart to the model, which corrected the colors and legend presentation.

A.3 Evaluating only the numerical extraction

	PolyCompChartIE						MetalThermoChartIE					
	Original Chart			Recreated Chart			Original Chart			Recreated Chart		
Model	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
o1	**50.31**	**54.84**	**51.63**	-	-	-	36.59	**30.90**	**31.12**	-	-	-
GPT-4o	44.71	37.16	38.42	50.08	41.14	42.88	36.39	17.32	20.81	**44.95**	21.93	25.40
(Models using Gemini for CSV extraction):												
LLaMa 3.1 Vision 11B + Finetune	52.39	51.81	51.93	-	-	-	**73.40**	50.62	**58.17**	-	-	-
LLaMa 3.1 Vision 11B	53.08	55.96	53.9	-	-	X	56.96	53.84	54.24	-	-	-
Qwen2.5-VL	**53.76**	**58.44**	55.38	-	-	-	56.97	**56.22**	55.51	-	-	-

Table 2: Precision (P), Recall (R), and F_1-score (F_1) for different models on the PolyCompChartIE and MetalThermoChartIE datasets, comparing original and recreated charts. Only x axis values and y axis values are evaluated.

A.4 Comparison of error bounds

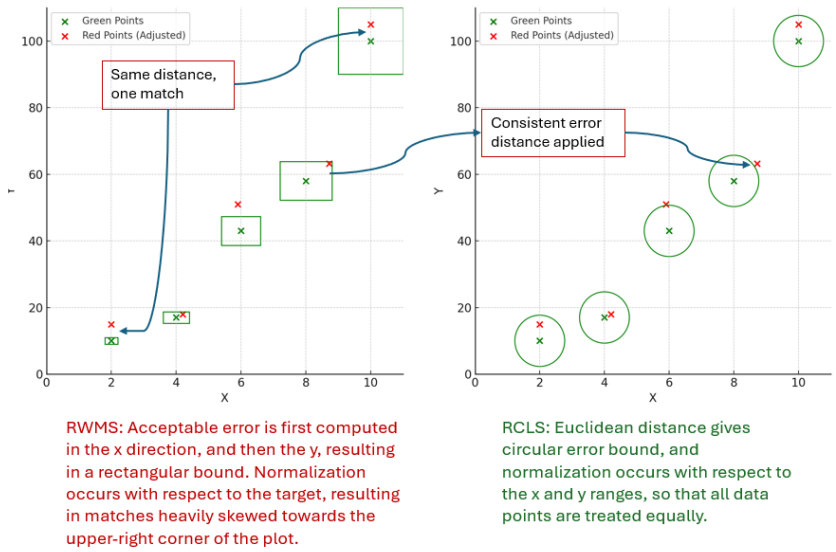


Figure 4: Comparison of error bounds between RWMS (left) and RCLS (right) metrics. RWMS applies rectangular error bounds with normalization relative to target values, resulting in inconsistent matching that favors points in the upper-right corner. RCLS implements circular error bounds using Euclidean distance with normalization based on axis ranges, ensuring consistent treatment of all data points regardless of their position on the plot.

A.5 Annotation protocol

The MetalThermoChartIE dataset was digitized over several months using Engauge Digitizer (https://digitizer.sourceforge.net), with each chart taking 10–60 minutes depending on complexity. Annotations include both axis-referenced and pixel-referenced coordinates, with every file independently reviewed and approved by a second person. We use the axis referenced annotations for our task.

The process of creating the PolyCompChartIE dataset was similar to that of the MetalThermoChartIE dataset. An online tool called WebPlotDigitizer Automeris (2024) was used. To use the tool, we manually entered the minimum and maximum values for the x and y axes, and clicked on their corresponding locations on the image of the figure. We then clicked on each of the datapoints, so that the tool would populate an excel file of the datapoints interpolated using the human provided bounds. Finally, we manually entered the legend label, if present in the figure, for each (x, y) coordinate. This process takes approximately 15 minutes per chart, representing a difficult bottleneck in the curation of highly complex and domain specific datasets where the underlying data is not available.

A.6 Finetuning hyperparameters

Data Usage

- 50 training steps, equivalent to approximately 5 shuffled epochs
- 372 training samples and 124 validation samples from MetalThermoChartIE
- The last 124 samples were held out as a test set
- Global batch size: 32
- Image preprocessing: `MllamaImageProcessor` defaults for normalization, padding, and rescaling

Optimization

- Optimizer: AdamW
- $\beta_1 = 0.9$, $\beta_2 = 0.999$
- Epsilon: 1×10^{-8}
- Gradient accumulators in FP32
- Learning rate: constant 5×10^{-5} with a 20-step warmup
- Weight decay: dynamic ($1 \times 10^{-2} \times$ learning rate)
- Gradient clipping: global norm of 1.0
- Mixed precision: FP16 forward pass, FP32 backward pass

Parameter-Efficient Fine-Tuning (PEFT with LoRA)

- LoRA rank: 16
- Alpha: 32
- Dropout: 0.05
- Target modules: Query, Key, Value, and Head-Mixing Output projections
- Bias: none (matching LLaMA-3 configuration)

Exploration Space

We also explored the following hyperparameter combinations:

- Epochs: $\{3, 5, 10\}$
- Batch size: $\{16, 32, 64\}$
- Learning rate: $\{5 \times 10^{-4}, 5 \times 10^{-5}\}$
- LoRA rank: $\{8, 16\}$

The best performance on validation data was obtained using 5 epochs, a batch size of 32, a learning rate of 5×10^{-5}, and a LoRA rank of 16.