A Hybrid LLM and Supervised Model Pipeline for Polymer Property Extraction from Tables in Scientific Literature

Van-Thuy Phi[1], Dinh-Truong Do[1,2*], Hoang-An Trieu[1,2*], Yuji Matsumoto[1]

[1] RIKEN Center for Advanced Intelligence Project
[2] Japan Advanced Institute of Science and Technology
{thuy.phi, truong.do, an.trieu, yuji.matsumoto}@riken.jp

Abstract

Extracting structured information from tables in scientific literature is a critical yet challenging task for building domain-specific knowledge bases. This paper addresses extraction of 5-ary polymer property tuples: (POLYMER, PROP_NAME, PROP_VALUE, CONDITION, CHAR_METHOD). We introduce and systematically compare two distinct methodologies: (1) a novel two-stage Hybrid Pipeline that first utilizes Large Language Models (LLMs) for table-to-text conversion, which is then processed by specialized text-based extraction models; and (2) an end-to-end Direct LLM Extraction approach. To evaluate these methods, we employ a systematic, domain-aligned evaluation setup based on the expert-curated PoLyInfo database. Our results demonstrate the clear superiority of the hybrid pipeline. When using Claude Sonnet 4.5 for the linearization stage, the pipeline achieves a score of 67.92% F1@PoLyInfo, significantly outperforming the best direct LLM extraction approach (Claude Sonnet 4.5 at 56.66%). This work establishes the effectiveness of a hybrid architecture that combines the generative strengths of LLMs with the precision of specialized supervised models for structured data extraction.

1 Introduction

The field of materials science, particularly polymer science, generates vast amounts of data published in scientific articles. This data, often embedded in tables, is crucial for developing new materials, training predictive models, and enabling data-driven discovery. Automated Information Extraction (IE) systems are essential for curating this knowledge into structured, machine-readable databases like PoLyInfo (Otsuka et al., 2011).

Recent studies by Phi et al. (2024) and Do et al. (2025) introduced a new corpus and developed a practical system for extracting polymer-related concepts and properties from unstructured text, demonstrating the high performance of supervised models like W2NER (Li et al., 2022) for Named Entity Recognition (NER) and ATLOP (Zhou et al., 2021) for Relation Extraction (RE) on their PolyNERE corpus. However, these models are inherently designed for plain text and cannot be directly applied to the semi-structured format of tables. Conversely, Large Language Models (LLMs) are adept at parsing diverse data formats but often lack the accuracy of fine-tuned models for domain-specific tasks.

This paper bridges this gap by investigating a hybrid approach that synergizes the strengths of both paradigms for the complex task of table extraction. Our primary contributions are:

- We propose a two-stage method that first leverages an LLM's structural understanding to convert table rows into natural language paragraphs. This linearized text is then processed by the advanced text-based IE system components identified by Phi et al. (2024) and Do et al. (2025).
- We systematically compare five advanced LLMs in both our hybrid pipeline and a direct end-to-end extraction approach using carefully engineered, task-specific prompts.

[*] These authors contributed equally.

Proceedings of the Third Workshop for Artificial Intelligence for Scientific Publications, pages 94–102

- We introduce a new PoLyInfo-based benchmark for evaluating property extraction from tables, providing near-comprehensive coverage (~66% of property names) of critical, standardized properties in the domain.
- Our results demonstrate that the hybrid pipeline significantly outperforms the direct LLM approach, establishing it as a more robust method for this task.

2 Related Work

Traditional neural approaches have achieved strong performance in domain-specific text extraction tasks. The W2NER architecture (Li et al., 2022) has shown particular effectiveness in capturing complex entity structures in scientific text—such as flat, overlapping, and discontinuous entities—commonly found in materials science literature, as demonstrated by Do et al. (2025). For relation extraction, ATLOP (Zhou et al., 2021) reformulates the task as entity-pair linking, delivering robust performance on specialized corpora like PolyNERE. Domain-adapted language models, such as MatSciBERT, have further improved results for materials science applications. However, these specialized models remain constrained to plain text input, limiting their direct applicability to tabular data.

Recent research has demonstrated the remarkable zero-shot and few-shot capabilities of LLMs for NER and RE. Most approaches attempt direct, end-to-end extraction, where the model is prompted to output structured data from a given input. However, this method forces a single model to handle multiple complex sub-tasks (parsing, entity recognition, etc.), which can lead to hallucinations or conversational outputs ill-suited for scientific data extraction (Kumar et al., 2025).

Converting tabular structures for LLM processing has emerged as a critical research area, with various serialization methods showing different effectiveness depending on table complexity. Recent work has shown that table linearization quality significantly impacts downstream extraction performance, though optimal strategies remain domain-dependent (Deng et al., 2024).

Our work bridges these areas by proposing a hybrid pipeline that leverages LLMs for table-to-text conversion while utilizing specialized supervised models for robust extraction, specifically addressing the gap in scientific table information extraction for polymer property data.

3 Methodology

The input for our system is a multi-modal prompt, combining a high-fidelity table image with its corresponding textual caption and footnotes. Table images are extracted directly from scientific documents using the MinerU parser (Wang et al., 2024). This image-based approach is motivated by Circi et al. (2024), who demonstrated that visual layout cues enable vision-enabled LLMs to more accurately extract complex relationships from scientific tables compared to text-only inputs.

We formalize the task as extracting a set of 5-ary property information tuples from a given scientific table. This formalization is grounded in the schema of the PoLyInfo database (Otsuka et al., 2011), the largest expert-curated database for polymers. The target is a set of tuples $T = \{t_1, t_2, ..., t_n\}$, where each tuple t_i consists of five key entity types:

- POLYMER: The name of the polymer material (e.g., "*polyethylene*", "*poly(p-diethynylbenzene)*").
- PROP_NAME: The name of the physical or chemical property being described (e.g., "*glass transition temperature*", "*density*").
- PROP_VALUE: The measured value of the property, typically including units (e.g., "*25 MPa*", "*1.097 g/cm3*").
- CONDITION: The experimental conditions under which the property was measured (e.g., "*at 25°C*", "*under nitrogen atmosphere*").
- CHAR_METHOD: The characterization technique or method used for the measurement (e.g., "*DSC*", "*tensile testing*").

These five types represent the core elements required to form a complete and usable entry in a materials science knowledge base. The primary challenge in the context of tables lies in correctly associating information that is structurally fragmented. The goal of our system is to accurately parse the combination of visual and textual information to compose a comprehensive set of valid 5-ary tuples. We compare two distinct approaches to solve this task.

3.1 Method 1: Hybrid LLM and Supervised Model Pipeline

This method decomposes the task into two sequential stages, leveraging the optimal model type for each sub-task.

Stage 1: LLM-based Table-to-Text Conversion: An LLM is given the multi-modal prompt (table image, caption, footnotes) and is instructed to act as a domain expert to linearize each data row into a descriptive paragraph. A novel aspect of our approach is the carefully engineered prompt (see Appendix A), which transforms the LLM into a specialized pre-processor for our supervised models. The prompt's key innovation is a conditional grouping strategy: it instructs the LLM to create separate, self-contained paragraphs for each material (POLYMER or its composite), and further subdivides these by CHAR_METHOD only if a method is explicitly stated. This hierarchical grouping is crucial as it prevents the ambiguous association of multiple properties with their respective measurement contexts—a common challenge for downstream relation extraction models.

Furthermore, by enforcing a strict, single-line output format and text normalization rules (e.g., "*T_g"* to "*Tg"*), the prompt ensures the generated text is a consistent and machine-readable intermediate representation, optimized for the models in the subsequent stage.

Stage 2: Supervised Text-based Tuple Extraction: The text generated from Stage 1 is then processed by a fixed, pre-trained text extraction system composed of supervised models trained on the PolyNERE corpus (Phi et al., 2024), selected based on their proven high performance.

We employ a W2NER model (Li et al., 2022), which is adept at handling the flat, overlapped, and discontinuous entity structures common in scientific text. This architecture is similar to that used in the PolyMinder system (Do et al., 2025). To further optimize for the materials science domain, we pair it with the MatBERT encoder (Walker et al., 2021).

We utilize the ATLOP model (Zhou et al., 2021), a choice validated by its strong performance in prior work (Phi et al., 2024; Do et al., 2025). To effectively capture the complex relationships present in the text, the model is paired with the powerful DeBERTa-v3-large encoder (He et al., 2020).

3.2 Method 2: Direct Tuple Extraction using LLMs

This approach follows a conventional end-to-end paradigm. The same multi-modal prompt is passed to a vision-enabled LLM. The prompt (see Appendix B) instructs the model to analyze the table's visual structure and associated text to directly output a list of all identifiable property tuples. To ensure a fair comparison, this prompt is also highly engineered with a similar set of detailed instructions and critical rules. This method relies entirely on the LLM's in-context reasoning to perform all sub-tasks simultaneously and serves as a direct baseline to evaluate the effectiveness of our hybrid pipeline.

4 Experiments

4.1 Datasets

The ground truth for our evaluation was constructed through a manual alignment process. We sourced curated polymer property data from the expert-driven PoLyInfo database (Otsuka et al., 2011) and mapped it to relevant content within a corpus of 37 tables from 29 scientific papers. Our final golden set comprises 293 property information tuples. Each tuple contains three essential entities (POLYMER, PROP_NAME, and PROP_VALUE), supplemented with optional CONDITION and CHAR_METHOD entities when available in the PoLyInfo entry. We confirmed that the 37 evaluation tables have no overlap with the PolyNERE training corpus, ensuring that supervised models in Stage 2 were tested on entirely unseen content.

Our analysis shows that the PoLyInfo-based golden annotations cover ~66% of all property names found across the evaluated tables. Specifically, we manually counted 132 property names appearing in the row and column headers of the 37 tables. The PoLyInfo database is an expert-curated resource where domain experts selectively extract and store only the most critical and standardized property information from scientific papers. Of the 132 property names in our tables, 87 (66%) have corresponding entries in PoLyInfo and were used to construct our golden set of 293 tuples. The remaining 45 property names (34%) may represent less critical properties that were not prioritized by expert curators for inclusion in PoLyInfo. Our evaluation is therefore near-comprehensive in its assessment of the most

important, standardized properties deemed critical by domain experts for polymer characterization.

For the hybrid pipeline, predicted binary relations from the ATLOP model are merged into 5-ary tuples based on the relation schema defined in Phi et al. (2024). During manual evaluation of these composed tuples (for both methods), we observed a consistent one-to-one mapping between a golden tuple and a corresponding prediction for each (POLYMER, PROP_NAME) pair (see Appendix E). A prediction is marked as True (T) only if all five of its constituent entities exactly match the golden tuple; otherwise, it is marked as False (F).

4.2 Results

Task	Model	Encoder	P	R	F1
NER	W2NER	MatBERT	**78.79**	**79.81**	**79.30**
	Baseline	MatSciBERT	78.05	76.53	77.28
RE	ATLOP	DeBERTa-v3-large	**87.93**	**86.89**	**87.40**
	Baseline	MatSciBERT	83.99	82.49	83.23

Table 1: NER and RE performance on the PolyNERE test set. RE uses gold entities.

Based on the observed one-to-one mapping in our evaluation setup, the number of False Positives and False Negatives are equivalent for the set of evaluated golden tuples. Consequently, Precision and Recall converge to the same value. We therefore report this unified metric as F1@PoLyInfo, representing the percentage of correctly extracted tuples from the set of important, PoLyInfo-defined properties:

*F1@PoLyInfo (%) = # True / (# True + # False) * 100*

We trained the supervised W2NER and ATLOP models using established hyperparameters from prior work (30 epochs, batch size 8, Adam optimizer). All LLM inferences were performed with deterministic settings (temperature=0, top_p=1).

We first establish the performance of our pipeline's core supervised models by evaluating them on the PolyNERE test set against the PolyMinder baseline (Do et al., 2025). Table 1 shows our selected models significantly outperform the established baseline for text-based extraction in this domain. Our W2NER+MatBERT configuration improves the NER F1 score by +2.02 points, while our ATLOP+DeBERTa-v3-large model shows a more significant +4.17 F1 point gain for RE. These results confirm their role as a powerful foundation for processing the linearized table data.

Model	Hybrid Pipeline			LLM Extraction		
	True	False	F1	True	False	F1
Claude Sonnet 4.5	199	94	**67.92**	166	127	**56.66**
GPT-4.1	112	181	38.23	119	174	40.61
GPT-4o mini	142	151	48.46	141	152	48.12
Gemini 2.5 Flash	164	129	55.97	73	220	24.91
Qwen2.5-VL 32B	158	135	53.92	86	207	29.35

Table 2: Model performance results.

We then evaluated the two end-to-end methodologies on our table extraction task. The results are summarized in Table 2.

The hybrid pipeline proves to be the superior strategy for the majority of the tested models. The advanced LLMs, Claude Sonnet 4.5, Gemini 2.5 Flash, and Qwen2.5-VL 32B Instruct, all saw dramatic performance increases when used in the hybrid pipeline. Specifically, Gemini 2.5 Flash and Qwen2.5-VL 32B improved by an absolute +31.06% and +24.57%, respectively, indicating that decomposing the complex task is critical for these models.

The best performance in our study was achieved by the hybrid pipeline, with Claude Sonnet 4.5 in the linearization stage reaching 67.92% F1@PoLyInfo. This represents a substantial +11.26% absolute improvement over its already strong direct extraction performance. An important exception to the general trend is GPT-4.1, for which the direct extraction method performed slightly better (40.61%) than the hybrid pipeline (38.23%). Similarly, the performance of GPT-4o mini was nearly identical across both methods. This suggests that for certain models, error propagation in a two-stage process—where suboptimal text generation in Stage 1 negatively impacts the supervised models—can outweigh the benefits of task decomposition. A detailed case study in Appendix D analyzes the specific failure modes of the pipeline for GPT-4.1.

The direct extraction method proved significantly more challenging for the majority of LLMs, with steep performance drops for models like Gemini 2.5 Flash and Qwen2.5-VL 32B highlighting the immense difficulty of simultaneously parsing a 2D structure and composing complex relations in a single step.

The hybrid pipeline's success lies in assigning the right task to the right model. The LLM excels at the generative, context-aware task of converting a table into fluent text. The supervised W2NER and ATLOP models, which are pre-trained and fine-tuned for their specific tasks, then excel at precise, closed-set extraction from this clean, textual input. This hybrid architecture proves more robust and accurate for most models, though it is not a universally guaranteed improvement, as seen with GPT 4.1.

5 Conclusion

In this work, we compared a hybrid pipeline (LLM linearization and supervised NER/RE) against a direct LLM approach for property extraction from tables, finding the hybrid architecture to be the more robust strategy on our PoLyInfo-based benchmark. Our best pipeline configuration achieves 67.92% F1@PoLyInfo, demonstrating that task decomposition with specialized supervised models yields superior performance compared to end-to-end LLM approaches.

Limitations

First, the evaluation set, while carefully curated, is of moderate size (293 tuples from 37 tables) and focused exclusively on the polymer science domain, and performance may vary on other types of scientific tables. Second, the hybrid pipeline's performance is highly dependent on the quality of the LLM-generated text in Stage 1, and as shown with GPT-4.1, poor linearization can create a bottleneck. Third, the success of our hybrid pipeline relies on the availability of well-trained text analyzers for NER and RE. This approach presupposes that high-quality, domain-specific supervised models are available for the second stage. Finally, our prompts were carefully designed with domain-specific instructions, but we did not systematically evaluate sensitivity to prompt variations. Evaluation requires manual normalization of tuples before matching, making comprehensive prompt experiments labor-intensive. Future work could explore automated evaluation methods for systematic prompting strategy comparison.

References

Circi, D., Khalighinejad, G., Chen, A., Dhingra, B., & Brinson, L. (2024). Extracting Materials Science Data from Scientific Tables. In *ACL 2024 Workshop Language+Molecules*.

Do, T. D., Trieu, A. H., Phi, V. T., Le Nguyen, M., & Matsumoto, Y. (2025, January). PolyMinder: A Support System for Entity Annotation and Relation Extraction in Polymer Science Documents. In *Proceedings of the 31st International Conference on Computational Linguistics: System Demonstrations* (pp. 1-8).

He, P., Liu, X., Gao, J., & Chen, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint* arXiv:2006.03654.

Kumar, P., Kabra, S., & Cole, J. M. (2025). MechBERT: Language Models for Extracting Chemical and Property Relationships about Mechanical Stress and Strain. *Journal of Chemical Information and Modeling*.

Li, J., Fei, H., Liu, J., Wu, S., Zhang, M., Teng, C., Ji, D. and Li, F., 2022, June. Unified named entity recognition as word-word relation classification. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 36, No. 10, pp. 10965-10973).

Naihao Deng, Zhenjie Sun, Ruiqi He, Aman Sikka, Yulong Chen, Lin Ma, Yue Zhang, and Rada Mihalcea. 2024. Tables as Texts or Images: Evaluating the Table Reasoning Ability of LLMs and MLLMs. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 407–426, Bangkok, Thailand. Association for Computational Linguistics.

Nicholas Walker, Amalie Trewartha, Haoyan Huo, Sanghoon Lee, Kevin Cruse, John Dagdelen, Alexander Dunn, Kristin Persson, Gerbrand Ceder, and Anubhav Jain. 2021. The impact of domain-specific pre-training on named entity recognition tasks in materials science. Available at SSRN 3950755.

Otsuka, S., Kuwajima, I., Hosoya, J., Xu, Y., & Yamazaki, M. (2011, September). PoLyInfo: Polymer database for polymeric materials design. In *2011 International Conference on Emerging Intelligent Data and Web Technologies* (pp. 22-29). IEEE.

Phi, V. T., Teranishi, H., Matsumoto, Y., Oka, H., & Ishii, M. (2024, May). PolyNERE: A novel ontology and corpus for named entity recognition and relation extraction in polymer science domain. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)* (pp. 12856-12866).

Wang, B., Xu, C., Zhao, X., Ouyang, L., Wu, F., Zhao, Z., Xu, R., Liu, K., Qu, Y., Shang, F. and Zhang, B., 2024. Mineru: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*.

Zhou, W., Huang, K., Ma, T., & Huang, J. (2021, May). Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 35, No. 16, pp. 14612-14620).

A Prompt for LLM-based Table-to-Text Conversion (Method 1)

You are analyzing a scientific table image. Convert it into structured natural language text that will be processed by Named Entity Recognition (NER) and Relation Extraction (RE) models.

TABLE CAPTION: [INSERT CAPTION TEXT HERE]
FOOTNOTES: [INSERT FOOTNOTES TEXT HERE]

TASK: Create separate paragraphs for each material to prevent entity confusion. If different properties are measured using different characterization methods (found in caption, footnotes, or column headers), create separate paragraphs for each material-method combination.

CRITICAL: Only separate by characterization method if methods are explicitly stated. If no methods are mentioned, write all properties for a material in one paragraph.

OUTPUT STRUCTURE:
1. First sentence: Introduce the table using the caption
2. Then, for EACH material:
*- **If characterization methods are specified**: Write separate paragraphs for each method*
*- **If NO methods are specified**: Write one paragraph with all properties*
3. Add blank line between paragraphs

REQUIREMENTS FOR EACH PARAGRAPH:
- Start with the material name EXACTLY as it appears in the table
*- **If characterization method is specified**: Include it after material name*
*- **If NO method is specified**: Omit method phrase entirely*
- List properties with their values and units
- Include any conditions from the caption, footnotes, or column headers
- Write each paragraph as a SINGLE continuous line
*- **Format with method**: "For [material name] measured by [CHAR_METHOD] [condition phrase]: [property name] is [value unit], [property name] is [value unit], ..."*
*- **Format without method**: "For [material name] [condition phrase]: [property name] is [value unit], [property name] is [value unit], ..."*

ENTITY TYPES TO INCLUDE:
1. POLYMER: Material/polymer name exactly as written in the table
2. PROP_NAME: Complete property name from column header
3. PROP_VALUE: Numerical value WITH unit (e.g., "7.29 MPa", "266.53%", "45.2 wt%")
4. CONDITION: Experimental conditions starting with a preposition (e.g., "at X°C", "with n=Y", "under annealing")
5. CHAR_METHOD: Measurement or characterization method as a noun phrase (e.g., "SEC", "DSC", "tensile testing")

CRITICAL RULES:
- Use material names EXACTLY as they appear in the table (no expansion or modification)
*- **DO NOT treat property names as characterization methods***
*- **Only use "measured by" when an actual measurement technique is specified (e.g., SEC, NMR, DSC, XRD, TEM, SEM, FTIR)***
*- **Column headers showing property names (e.g., "Tensile strength", "Density", "Modulus") are NOT characterization methods***
- Separate by characterization method only when methods are explicitly mentioned
- Copy exact numbers and units from the table
- Include units WITH values (e.g., "7.29 MPa" not just "7.29")
- Each paragraph must be a single continuous line - NO line breaks within a paragraph
- Add blank line between paragraphs only
- DO NOT use subscript notation with underscores (e.g., M_n, T_g, T_c). Instead use simplified notation (e.g., Mn, Tg, Tc)
- Condition phrases must start with a preposition (e.g., "at", "under", "with", "in", "by")
- CHAR_METHOD must be a noun phrase (e.g., "DSC", "tensile testing", "X-ray diffraction")

EXAMPLE FORMAT:

Case 1 - NO characterization methods specified:
This table presents [property category] of [material type] materials.

For [Material-A] [condition phrase if any]: [property-1] is [X.XX unit], [property-2] is [Y.YY unit], [property-3] is [Z.ZZ unit].

For [Material-B] [condition phrase if any]: [property-1] is [X.XX unit], [property-2] is [Y.YY unit], [property-3] is [Z.ZZ unit].

Case 2 - Characterization methods ARE specified:
This table presents [property category] of [material type] materials.

For [Material-A] measured by [CharMethod-1]: [property-1] is [X.XX unit], [property-2] is [Y.YY unit].

For [Material-A] measured by [CharMethod-2] [condition phrase if any]: [property-3] is [Z.ZZ unit], [property-4] is [W.WW unit].

For [Material-B] measured by [CharMethod-1]: [property-1] is [X.XX unit], [property-2] is [Y.YY unit].

For [Material-B] measured by [CharMethod-2] [condition phrase if any]: [property-3] is [Z.ZZ unit], [property-4] is [W.WW unit].

OUTPUT: Return ONLY the converted text. No explanations or additional commentary.

B Prompt for Direct Tuple Extraction (Method 2)

You are analyzing a scientific table image. Extract ALL property measurements from the table as structured tuples.

TABLE CAPTION: [INSERT CAPTION TEXT HERE]
FOOTNOTES: [INSERT FOOTNOTES TEXT HERE]

TASK: Extract ALL property measurements from the table as 5-element tuples.

TUPLE FORMAT:
(POLYMER, PROP_NAME, PROP_VALUE, CONDITION, CHAR_METHOD)

REQUIREMENTS FOR EACH TUPLE:
- Extract one tuple for EACH property measurement (one row × one column = one tuple)
- Include the complete material name in every tuple
- Copy exact values with units from table cells
- Extract any conditions or methods from the caption, footnotes, or column headers
- Process systematically: for each material (row), extract all properties (columns)

ENTITY TYPES TO INCLUDE:
1. POLYMER: Material/polymer name exactly as written in the table (e.g., "PE", "Sample A", "Composite-5")
2. PROP_NAME: Complete property name from column header (e.g., "tensile strength", "glass transition temperature")
3. PROP_VALUE: Numerical value WITH unit (e.g., "X.XX MPa", "YY.Y%", "Z.ZZ ± 0.XX unit")
4. CONDITION: Experimental conditions starting with a preposition (e.g., "at X°C", "with n=Y", "under annealing", "in air")
5. CHAR_METHOD: Measurement or characterization method as a noun phrase (e.g., "tensile testing", "thermal analysis", "SEC", "DSC")

CRITICAL RULES:
- Use material names EXACTLY as they appear in the table (no expansion or abbreviations)
- Repeat material names in every tuple for clarity
- Copy exact numbers and units from the table
- Include units WITH values (e.g., "7.29 MPa" not just "7.29")
- Extract conditions/methods from caption, footnotes, and headers
- CONDITION must start with a preposition (e.g., "at", "under", "with", "in", "by")
- CHAR_METHOD must be a noun phrase (e.g., "DSC", "tensile testing", "X-ray diffraction")
- If condition or method not specified, use empty string ""
- One measurement = one tuple
- DO NOT use subscript notation with underscores (e.g., M_n, T_g, T_c). Instead use simplified notation (e.g., Mn, Tg, Tc)

EXAMPLE FORMAT (using placeholder values):
("PE", "property 1", "value unit", "at condition", "method name")
("PE", "property 2", "value unit", "", "")
("Sample C", "property 1", "value ± error unit", "at condition 1, with condition 2", "characterization method")

OUTPUT: Return ONLY the tuple list. One tuple per line. No explanations or additional commentary.

C Examples of Evaluated Tables

Table 1. Molecular Characteristics of CE Copolymers

polymer	M_n[a] (kg/mol)	M_w/M_n[a]	cyclohexyl ethylene[b] (wt %)	ethyl branches per 100 backbone carbon atoms in ethylene units[b]	T_g[c] (°C)	ρ[d] (g/cm^3)
PE	65	1.04		3.5	−31	0.907
CE50	49	1.05	51	3.5	−20	0.918
CE60	67	1.05	60	2.6	7	0.928
CE70	52	1.05	72	3.3	30	0.937
CE80	65	1.05	84	2.4	43	0.947
PCHE	70	1.05	100		144	0.960

[a]Measured with SEC using the parent SB copolymer with universal calibration. The Mark−Houwink parameters for PS and PB are K_{PS} = 8.63 × 10^{-3} mL/g, α_{PS} = 0.736 and K_{PB} = 25.2 × 10^{-3} mL/g, α_{PB} = 0.727.[33] The K and α of SB copolymers are estimated using the weight-averaged values of the homopolymer counterparts. [b]Calculated from the integration of characteristic peaks in ^{1}H NMR spectra. [c]Determined with DSC. [d]Measured with density gradient column at 23 °C.

Figure 1: Example table from the evaluation set, featuring complex headers and footnotes linking properties to characterization methods (SEC, DSC, NMR).

Table 1. Characteristics of Both PFS and PWN2010

	GPC			IEC [mmol g^{-1}]		TGA	DSC	water uptake[c]	
polymer	M_n	M_w	PD	calcd	found	T_{decomp} [°C]	T_g [°C]	[wt %]	[λ][d]
PFS	30 200	59 000[a]	1.92	0		285	105.5	0	0
PWN2010	9 000	67 000	7.5	7.8[b]	7.0	340	>330	18	2.5

[a] Value is received from the supplier (Monomer Polymer & Dajac Laboratories, USA). [b] Calculation is based on 100% substitution (1 $-PO_3H_2$/aromatic ring). [c] Value at RH = 50%, T = 30 °C. [d] $\lambda = [H_2O]/[-PO_3H_2]$.

Figure 3: Input table for the error analysis.

Table 1. Thermal properties of LPEEK/HPEEK blends with various HPEEK contents

LPEEK/HPEEK (w/w)	T_g (°C)[a]	T_g (°C)[b]	T_c (°C)	T_m (°C)	X_c (%)	T_d[5] (°C)
100/0	145	167	283	334	15.4	560
99/1	145	170	288	337	15.5	566
98/2	145	171	291	337	17.2	570
97/3	144	172	292	338	17.2	570
96/4	144	173	293	338	17.6	570
95/5	144	174	294	338	19.5	565

[a] Measured using DSC.
[b] Measured using DMA.

Figure 2: Example table illustrating a common ambiguity where two columns share a property name (Tg) but are distinguished by different characterization methods (DSC, DMA) in the footnotes.

D Error Analysis: Understanding the GPT-4.1 Performance Anomaly

As noted in Section 4.2, GPT-4.1 was a notable exception to the general trend, performing slightly better with direct extraction than within the hybrid pipeline. This appendix provides a case study to dissect the specific failure modes that explain this result, highlighting the critical role of the table-to-text linearization stage and the potential for error propagation. We use the table in Figure 3 and its corresponding PoLyInfo-based golden annotations in Figure 4 as a running example.

Analysis of the Hybrid Pipeline (Method 1): In the hybrid pipeline, the quality of the final extraction is entirely dependent on the quality of the text generated by the LLM in Stage 1. Figure 3 shows a table where characterization methods (TGA, DSC) are specified as column headers.

Generated Text from GPT-4.1: For the input table in Figure 3, GPT-4.1 produced the following linearized text for Stage 2:

> *"This table presents characteristics of both PFS and PWN2010.*
> *For PFS at RH = 50%, T = 30 °C: Mn is 30 200, Mw is 59 000, PD is 1.92, IEC calcd is 0 mmol g^{-1}, IEC found is 0 mmol g^{-1}, Tdecomp is 285 °C, Tg is 105.5 °C, water uptake [wt %] is 0, water uptake [λ] is 0.*
> *For PWN2010 at RH = 50%, T = 30 °C: Mn is 9 000, Mw is 67 000, PD is 7.5, IEC calcd is 7.8 mmol g^{-1}, IEC found is 7.0 mmol g^{-1}, Tdecomp is 340 °C, Tg is >330 °C, water uptake [wt %] is 18, water uptake [λ] is 2.5."*

This generated text contains two critical flaws that lead to downstream extraction failures:

- **Omission of Characterization Methods:** The LLM failed to identify "*TGA*" and "*DSC*" from the column headers as characterization methods and did not include them in the generated paragraphs. Because this information is completely absent from the text, it is impossible for the downstream supervised W2NER and ATLOP models to extract the CHAR_METHOD entity. This results in an immediate and unavoidable False evaluation for four of the six golden tuples shown in Figure 4.

- **Incorrect Value-Unit Representation**: The linearization format "*...water uptake [wt %] is 0...*" separates the property's unit from its value. The supervised NER model, which relies on surface text patterns, struggles with this structure. It is likely to identify PROP_VALUE as just "*0*" and incorrectly associate "*[wt %]*" with the PROP_NAME. This creates a mismatch with the golden annotation in Figure 4, which correctly defines PROP_NAME as "*water uptake*" and PROP_VALUE as "*0 wt%*".

These linearization errors propagate through the pipeline, preventing the supervised models in Stage 2 from performing correctly and resulting in a low overall score.

POLYMER	PFS
PROP_NAME	Tg
PROP_VALUE	105.5 ◦C
CONDITION	
CHAR_METHOD	DSC
POLYMER	PFS
PROP_NAME	water uptake
PROP_VALUE	0 wt%
CONDITION	at RH = 50%, T = 30 °C
CHAR_METHOD	
POLYMER	PFS
PROP_NAME	Tdecomp | Thermal decomposition temperature
PROP_VALUE	285 ◦C
CONDITION	
CHAR_METHOD	TGA
POLYMER	PWN2010
PROP_NAME	Tg
PROP_VALUE	>330 ◦C
CONDITION	
CHAR_METHOD	DSC
POLYMER	PWN2010
PROP_NAME	water uptake
PROP_VALUE	18 wt%
CONDITION	at RH = 50%, T = 30 °C
CHAR_METHOD	
POLYMER	PWN2010
PROP_NAME	Tdecomp | Thermal decomposition temperature
PROP_VALUE	340 ◦C
CONDITION	
CHAR_METHOD	TGA

Figure 4: The PoLyInfo-based golden annotations for the table in Figure 3. These tuples serve as the ground truth for the error analysis, highlighting failures in CHAR_METHOD and PROP_VALUE extraction.

Analysis of Direct LLM Extraction (Method 2): In the direct extraction method, the LLM is responsible for parsing the table and generating tuples in one step. Below are the corresponding outputs from GPT-4.1 for the golden tuples in Figure 4. Incorrectly predicted entities are shown in **bold**.

("PFS", "Tdecomp", "285 °C", "", "TGA")
("PFS", "Tg", "105.5 °C", "", "DSC")
*("PFS", "**water uptake [wt %]**", "**0**", "at RH = 50%, at T = 30 °C", "")*
("PWN2010", "Tdecomp", "340 °C", "", "TGA")
("PWN2010", "Tg", ">330 °C", "", "DSC")
*("PWN2010", "**water uptake [wt %]**", "**18**", "at RH = 50%, at T = 30 °C", "")*

From these outputs, we observe:

- **Correct CHAR_METHOD Extraction**: For properties with simple headers like "*Tdecomp*" and "*Tg*", the direct method performs perfectly, correctly identifying "*TGA*" and "*DSC*" as the CHAR_METHOD. This gives it an advantage over the flawed pipeline output, where this information was lost.
- **Incorrect PROP_NAME and PROP_VALUE Parsing**: Similar to the pipeline's issue, the direct method also struggles with the complex "*water uptake*" header. It incorrectly merges the unit "*[wt %]*" into the PROP_NAME and extracts only the numerical part ("*0*" or "*18*") as the PROP_VALUE, leading to a mismatch.

This case study explains the GPT-4.1 performance anomaly. The hybrid pipeline's linearization stage made a significant error by omitting CHAR_METHOD information, leading to unavoidable downstream failures for the supervised models. In contrast, the direct extraction method, while also imperfect, correctly extracted more of the golden tuples. This demonstrates the risk of error propagation in a pipeline. If an LLM's text generation style is a poor fit for the downstream models, a direct approach can, in some cases, yield slightly better results by avoiding this cascade of errors.

E One-to-One Mapping in Tuple Evaluation

We observed a consistent one-to-one mapping between golden tuples and predictions for each (POLYMER, PROP_NAME) pair across all evaluated tuples.

For the Hybrid Pipeline: The ATLOP model predicts binary relations that are merged into 5-ary tuples following Phi et al. (2024). When multiple binary relations share the same (POLYMER, PROP_NAME, PROP_VALUE) triple, they are consolidated into a single tuple by aggregating CONDITION and CHAR_METHOD entities.

For Direct LLM Extraction: Scientific tables organize data with one measurement per cell. The prompt instructs "*Extract one tuple for EACH property measurement (one row × one column = one tuple)*", and all LLMs followed this instruction.

Under this one-to-one constraint, each incorrect prediction simultaneously represents both a false positive and a false negative, making these counts equivalent.