SudokuFill: A Multi-Agent Progressive Filling Framework for Document-Level Scientific Information Extraction

Yang Li[1,3] **Yajiao Wang**[1,3] **Yu Zhang**[2] **Yuanzhe Zhang**[1,3]
Maodi Hu[1,3*] **Mengting Zhang**[1,3] **Xi Sun**[1] **Hua Yue**[2*] **Zhixiong Zhang**[1,3*]

[1]National Science Library, Chinese Academy of Sciences
[2]Institute of Process Engineering, Chinese Academy of Sciences
[3]Department of Information Resources Management, School of Economics and Management, University of Chinese Academy of Sciences

Abstract

Scientific information extraction (SciIE) is a key bottleneck for turning unstructured papers into computable knowledge bases, yet most existing systems still follow a "local extraction then global assembly" paradigm. This workflow is inherently lossy: by extracting fields in isolation, it breaks global correlations and discards high-confidence signals that could otherwise be reused as internal supervision, forcing systems to repeatedly restart from scratch, especially in long, multimodal scientific documents. In this paper, we propose a different view: SciIE should be solved as a progressive filling problem, similar to solving a Sudoku, once a field is filled with high confidence, it should act as a constraint that guides the remaining uncertain fields. Based on this idea, we introduce **SudokuFill**, a multi-agent framework that maintains a Global Filling State and performs priority scheduling to establish reliable anchors first, then reuses them as internal supervision for iterative deliberation over harder fields. Evaluated on a specialized document-level adjuvant dataset, our framework achieves a SOTA score of 51.83% on our benchmark. Crucially, **SudokuFill** enables a 7B model to outperform the vanilla GPT-4o, suggesting that structured architectural reasoning can effectively compensate for parameter scale.

1 Introduction

With the burgeoning AI for Science (AI4S) paradigm, high-quality data has emerged as the indispensable foundation driving scientific discovery (Dagdelen et al., 2024; Sun et al., 2025). However, a vast amount of domain knowledge remains sedimented in an unstructured format across hundreds of millions of scientific papers, with its core value often encapsulated in the fine-grained descriptions of specific Research Objects and their Complete Attributes. This unstructured data modality directly creates a bottleneck, obstructing the direct conversion of massive literature archives into AI training data and thereby constraining the potential of AI models in frontier tasks such as mechanism discovery and hypothesis generation (Zhou et al., 2024). Consequently, developing a document-level SciIE framework to facilitate the construction of computable knowledge bases has become a foundational imperative within the AI4S community.

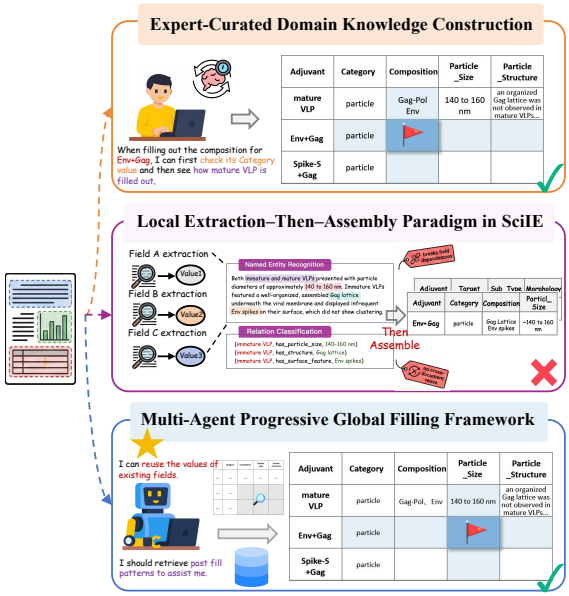

Figure 1: Comparison between the conventional "local extraction then global assembly" paradigm (**middle**) and our expert-inspired database population view (**top**), instantiated as a multi-agent progressive global filling framework (**bottom**).

Historically, limited by early model capacities, scientific database construction has predominantly relied on a "local extraction then global assembly" paradigm (Liu et al., 2021). Prior systems began with a target schema, decomposed it into isolated, field-level information extraction (IE) subtasks, extracted candidate values from sentence or paragraph level contexts, and then assembled the

*Corresponding authors: Maodi Hu (humaodi@mail.las.ac.cn), Hua Yue (hyue@ipe.ac.cn), and Zhixiong Zhang (zhangzhx@mail.las.ac.cn).

Findings of the Association for Computational Linguistics: ACL 2026, pages 33112–33138
July 2-7, 2026

field-wise predictions one by one into structured records via post-processing. However, we contend that this process is inherently lossy for SciIE, as it artificially severs vital global correlations. Specifically: **(i) scientific attributes are bound by intrinsic dependencies;** in protein functional annotation, for instance, determining a domain type imposes soft constraints that narrow the search space and mitigate ambiguity for subsequent fields like active sites (Dou et al., 2024). **(ii) High-confidence records from previous extractions naturally accumulate across documents, providing valuable internal supervision that should serve as rich referential guidance for subsequent extraction.** By neglecting these internal signals, systems are forced to restart from scratch for every field and document, increasing the risk of errors when processing information-dense, long-form text. Unfortunately, even as Large Language Models (LLMs) demonstrate strong reasoning and long-context capabilities, much of the SciIE studies persist in this fragmented paradigm (Schilling-Wilhelmi et al., 2025; Dagdelen et al., 2024), treating LLMs primarily as more efficient local extractors rather than leveraging them for global, structured inference.

In light of this, we reformulate document-level SciIE as a progressive, Sudoku-style filling problem over an evolving global knowledge state. We propose **SudokuFill**, a multi-agent, multi-round framework that explicitly models field dependencies and schedules extractions in an easy-to-hard order. High-confidence field predictions are iteratively written back to a global filling state and reused as structured constraints for subsequent queries, while extracted records are archived to provide cumulative reference patterns across documents. Through role specialization and multi-agent debate, the framework decomposes global reasoning into localized, precise extraction steps, improving consistency in long, multimodal documents.

We selected vaccine adjuvants (Singh and O'Hagan, 1999) as our target domain, which has long lacked a systematic database. Given the structural complexity of its research objects and attribute descriptions, we constructed the first document-level adjuvant benchmark to evaluate the specific task challenges and the framework proposed in this study. Unlike most mainstream SciIE benchmarks that provide manually cleaned and pre-segmented target paragraphs, we focus on the end-to-end process of knowledge base construction by taking raw PDFs as input. This setting requires the system to operate directly over complex layouts, cross-page evidence dispersion, and multi-source signals such as tables and figures, while identifying research objects and reconstructing their attribute sets at the document level. Overall, our main contributions are summarized as follows:

(1) To the best of our knowledge, we are the first to reframe document-level SciIE as a Sudoku-style filling, addressing the information loss of the prevailing "local extraction then global assembly" paradigm in long, multimodal scientific documents.

(2) We propose SudokuFill, a two-stage multi-agent framework with a Global Filling State that iteratively reuses extracted fields as constraints, enabling a 7B model to outperform vanilla GPT-4o.

(3) We contribute the first document-level vaccine adjuvant IE benchmark for SciIE evaluation. Extensive experiments on this benchmark reveal insights.

2 Related Work

We review the methodological evolution of mining structured knowledge from scientific literature, covering the transition from early heuristic systems and pre-trained models, such as BERT (Devlin et al., 2019) and T5 (Raffel et al., 2020), to generative LLMs, and finally to the latest agents and knowledge-enhanced frameworks.

Heuristic Systems and statistical rules. Early research in SciIE primarily relied on heuristic systems and statistical rules (Espinosa-Anke and Saggion, 2014; Storrer and Wellinghoff, 2006; Berlin and Motro, 2002), utilizing regular expressions, dictionary matching, and statistical metrics, such as DC-Value (Liwei, 2022) and entropy (Tian et al., 2023), to filter terms based on lexical rigidity. In the material science domain, the tool ChemDataExtractor (Swain and Cole, 2016) parsed chemical properties using dictionary-based rules. Beyond this, predefined templates, such as Hearst patterns, were utilized to infer semantic relations like hyponymy and synonymy (Liu et al., 2017). While effective for specific, rule-governed tasks, these methods struggle with diverse semantic expressions and require feature engineering.

Pre-trained Models. To transcend the limitations of shallow pattern matching, the field shifted towards Deep Learning, where pre-trained models emerged as the mainstream paradigm to capture semantic context. Representative studies anchored on BERT and its variants (Jain et al., 2023; Zhang

et al., 2023; Pérez-Pérez et al., 2022) have demonstrated significant superiority over traditional systems relying on handcrafted features across diverse scientific extraction tasks. For instance, multi-stage systems such as BERT-PSIE (Gilligan et al., 2023) integrate sentence filtering, named entity recognition, and relation classification into traceable pipelines, achieving high-precision attribute extraction within the materials science domain. Subsequently, strategies involving domain-adaptive pretraining (Gupta et al., 2022; Shetty et al., 2023) have further extended extraction capabilities across specific scientific disciplines (Beltagy et al., 2019; Lee et al., 2020). Beyond encoder-only architectures, Text + Chem T5 (Christofidellis et al., 2023) represents a paradigm shift, by leveraging multitask pre-training on 2.3 million reactant-product pairs, it explores generative approaches to chemical IE distinct from the BERT framework. Despite these continuous advancements, pre-trained models face inherent constraints: finite context windows (Beltagy et al., 2020) and the high cost of manual annotation (Li et al., 2024) severely limit their capacity to extract complex information from long scientific documents.

Generative LLMs. Leveraging extended context windows and reduced reliance on explicit annotation, LLMs have streamlined scientific IE by enabling low-cost, efficient extraction via prompt engineering and few-shot learning (Dagdelen et al., 2024; Zhang et al., 2024). Notable implementations include the ChemPrompt strategy (Zheng et al., 2023), which extracts Metal-Organic Framework (MOF) data from enriched text segments, and the foundation model nach0 (Livne et al., 2024), which integrates chemical and linguistic knowledge to solve complex mining tasks. While LLMs extend the context window beyond previous models, their reliability in extracting complex information from long scientific documents remains constrained by prone-to-error hallucinations (Dagdelen et al., 2024) and the "Lost in the Middle" phenomenon (Liu et al., 2024b).

Agents and Knowledge-Enhanced Frameworks. In materials science, Eunomia (Ansari and Moosavi, 2024) employs a chain-of-verification mechanism to ensure high-fidelity extraction of structured data. In catalysis, CATDA (Chen et al., 2025a) leverages text-to-graph construction to capture complex "synthesis-performance" relationships scattered across lengthy documents. Similarly, in clinical medicine, CLEAR (Lopez et al., 2025) significantly improves accuracy by substituting broad embedding search with precise entity-centric retrieval. However, existing frameworks remain imperfect in handling holistic long-document multimodal reasoning. Furthermore, they largely rely on static knowledge bases, lacking the self-evolutionary capability to dynamically refine extraction logic for complex field dependencies.

3 Proposed Method

3.1 Overview

As shown in Fig 2, we propose a round-driven, two-stage framework **SudokuFill** for structured extraction. Stage I schedules field-grounded queries by performing page-level probing and ranking them by extraction priority (§ 3.2). Stage II processes queries sequentially, resolving each via multi-round deliberation among heterogeneous agents to refine candidates until convergence (§ 3.3).

Crucially, the framework centers on cross-round and cross-query information reuse. Each round updates a history memory, while high-confidence converged results provide dynamic context to constrain subsequent queries. After processing each paper, extracted records are archived in a searchable global filling state, accumulating cross-document priors and patterns to support the row/column constraint agents. For clarity, Fig 2 depicts the workflow for a single query within a single round; in practice, global convergence arises from iterating over multiple queries across rounds.

3.2 Stage I: Field Priority Scheduling

Before entering multi-agent extraction, we introduce Stage I as a field-grounded query priority scheduling stage. Rather than targeting the final correctness of candidate values, Stage I performs a probing pass to determine whether each field-grounded query exhibits identifiable signals on document pages and to estimate the confidence strength of such signals, thereby providing an easy-to-hard execution order and a stable starting point for subsequent extraction. The pseudocode for the Stage I algorithm is provided in the appendix A.5.

Formally, we first instantiate each schema field f into a query unit $q = \langle f,\ \phi(f) \rangle$, where $\phi(f)$ specifies the field description and extraction constraints. This normalization improves agents' semantic comprehension with the target and facilitates reusing resolved query results as standard-

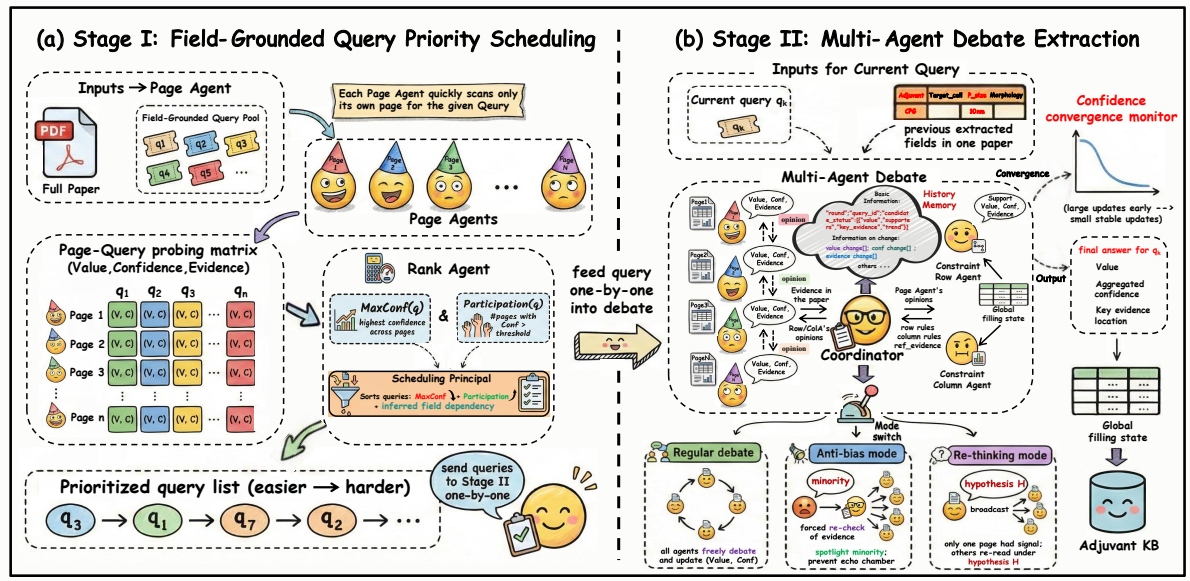

Figure 2: **Two-stage multi-agent framework for SciIE:** Stage I prioritizes queries via page agent using confidence and participation signals. Stage II extracts each query through multi-agent debate with coordination, mode switching, and confidence-based convergence, producing the final answer for each query and incrementally updates a **global filling state** for iterative reuse across subsequent queries and papers.

ized context in later rounds (the generation rules and templated mappings are provided in the appendix A.1). We build MLLM-based Page Agents, and feed these queries as inputs to perform a parallel, one-pass page-level probing over the document. Each Page Agent quickly scans only its assigned page for a given query and output a candidate triple $(V_{p,q}, C_{p,q}, E_{p,q})$ for each page p – query q pair, where V is the proposed value, C is the agent's internal confidence estimate (with a calibration rubric in the appendix A.2), and E provides verifiable evidence localization. We organize all page – query outputs into a page–query matrix $\mathcal{M}$ (Fig 2(a)), which serves as the direct input to the rank agent to characterize signal strength and spatial distribution across pages. Moreover, Candidates (V, C, E) in $\mathcal{M}$ are also directly injected into Stage II as the initial candidate pool, providing a warm-start for subsequent multi-round deliberation.

Given $\mathcal{M}$, the query execution order is determined by a rank agent. The rank agent considers three types of signals: (i) $\mathrm{MaxConf}(q) = \max_p C_{p,q}$, which takes the maximum confidence across pages for each query, used as a proxy for signal strength and to prioritize queries accordingly; (ii) participation, defined as $\mathrm{Part}(q) = |\{p \mid C_{p,q} > 0.5\}|$, where $C > 0.5$ indicates an effective participation to filter noisy responses driven by weak cues or uncertain matches (the threshold rationale and empirical analysis are deferred to the Appendix A.2 and Section 5.2); and (iii) schema-implied inter-field dependencies, which help prioritize queries that are more likely to provide constraints for subsequent fields. The rank agent outputs a prioritized query sequence, which is then executed in Stage II in order. (Prompts and decision specifications are detailed in the Appendix A.4.2)

3.3 Stage II: Multi-Agent Debate for Query Extraction

Stage II processes queries sequentially under the schedule π and resolves each via multi-round deliberation among heterogeneous agents. For each query $q \in \pi$, we initialize its candidate pool $\mathcal{C}^{(0)}(q)$ with warm-start candidates collected from the page–query matrix $\mathcal{M}$, the set of $(V_{p,q}, C_{p,q}, E_{p,q})$ across pages. The deliberation then proceeds in rounds, where agents exchange grounded evidence and structured constraints to iteratively refine candidate values until convergence.

Agents and global filling state. We employ three roles with explicitly constrained interfaces. *Page agents* $\mathcal{A}_P$ ground on their assigned pages and propose or revise candidates for the current query, outputting (V, C, E). *Constraint agents* provide complementary structural signals from a row view $\mathcal{A}_{\text{row}}$ and a column view $\mathcal{A}_{\text{col}}$. Both consult an evolving global filling state $\mathcal{G}$, a Sudoku-style state that stores high-confidence confirma-

tions from previously resolved queries. Importantly, constraint agents do not introduce new values; instead, they output (V^{sup}, C, E) with $V^{\text{sup}} \in \{ V \mid \exists C, E \text{ s.t. } (V, C, E) \in \mathcal{C}^{(t)}(q) \}$, thereby supporting or challenging candidates proposed by page agents. Finally, a *coordinator* $\mathcal{A}_C$ orchestrates the deliberation without directly predicting values. Throughout Stage II, we maintain the query-specific candidate pool $\mathcal{C}^{(t)}(q)$, a round history memory $H^{(t)}(q)$ that summarizes agents' stances and evidence, and the global filling state $\mathcal{G}$, which is incrementally updated after each query converges.

Round protocol with cross-round and cross-query reuse. At round t, the coordinator $\mathcal{A}_C$ builds the round context from the current query q, the candidate pool $\mathcal{C}^{(t)}(q)$, the compressed history $H^{(t-1)}(q)$, and the current global filling state $\mathcal{G}$. All agents then respond in parallel. Constraint agents consult $\mathcal{G}$ to check consistency and surface relevant constraints, and output (V^{sup}, C, E) restricted to current candidates. The coordinator aggregates all outputs to update the round memory $H^{(t)}(q)$ and the candidate pool $\mathcal{C}^{(t+1)}(q)$ by consolidating evidence, tracking support versus opposition, and recording within-agent confidence revisions. This yields reuse at two levels: cross-round reuse via $H^{(t)}(q)$ for subsequent rounds of the same query, and cross-query reuse by writing converged results back into $\mathcal{G}$ as standardized context for later queries.

Adaptive deliberation and convergence. Multi-agent deliberation may prematurely collapse to a majority view when evidence is sparse or unevenly distributed, leaving critical dissent insufficiently examined. To mitigate this, the coordinator $\mathcal{A}_C$ selects among three deliberation modes conditioned on the evolving history $H^{(t)}(q)$. **Regular debate** is the default, where page and constraint agents exchange evidence to refine the candidate pool. **Anti-bias debate** is activated when a dominant candidate is repeatedly endorsed while a persistent, evidence-backed objection remains; the coordinator then prioritizes directly addressing it. **Rethinking** is used when the leading candidate is supported by only a single page source or a single agent, prompting the coordinator to elicit confirmations or counter-evidence from other page agents. Convergence is judged by temporal trends rather than cross-agent confidence comparability. The coordinator stops when the leading candidate is stable across successive rounds and each agent's *within-agent* confidence updates become negligible, indicating diminishing revisions. It then outputs the final value V with aggregated evidence E and writes (q, V, E) into the global filling state $\mathcal{G}$ to constrain subsequent queries. (For more details of stage II and specific data flow for each round, see the Appendix B)

4 Experiments

In this section, we conduct a comprehensive evaluation of **SudokuFill**. Section 4.1 describes the experimental setup, Section 4.2 reports the main results, and Section 4.4 presents ablation studies analyzing the contributions of key components.

4.1 Experiment Setup

Dataset We evaluate **SudokuFill** on the Vaccine Adjuvant Benchmark, a document-level SciIE dataset designed for the end-to-end transition from raw literature to structured records. The benchmark comprises 250 scientific papers and over 1,000 annotated adjuvant records. Unlike traditional datasets, it requires identifying multiple research objects within a single document and populating a schema of 10 heterogeneous fields: Adjuvant_Name, Category, Sub_type, Composition, Morphology, Particle_Size, Particle_Structure, Target, Target_Cell and Combination_mode (details in Appendix C) This setting necessitates cross-page evidence synthesis and multi-source signal integration from raw PDFs. To ensure high fidelity, domain experts annotated the dataset in a double-blind process with arbitration, yielding robust ground truth for entity- and record-level consistency evaluation.

Automatic Evaluation Metrics To assess extraction performance, we employ three levels of metrics: (1) **Entity-level Metrics**: We report Precision (P), Recall (R), and Micro F1-score (Goutte and Gaussier, 2005) to evaluate the system's ability to extract individual attribute values correctly. This reflects the local precision of the agents. (2) **Row-level Metrics**: Given the Sudoku-style nature of the task, the coherence of an entire record is paramount. We introduce *Row-level Accuracy* and *Micro-F1*, which require the system to not only identify the research object (Adjuvant Name) but also correctly associate it with its 10 corresponding attributes. A row is considered a candidate for accuracy only if the core attributes are correctly grouped, providing a stringent measure of structural consistency. (3)

Model / Method	Params	Entity-level			Row-level		Overall
		P	R	F1	Acc	F1	Avg F1
Closed-source Multimodal Large Language Models							
GPT-4o (Achiam et al., 2023)	–	**69.50**	68.53	69.01	31.42	26.43	47.72
GPT-4o mini (OpenAI, 2024b)	–	65.19	68.37	66.74	29.81	26.07	46.41
GPT-5 Nano (OpenAI, 2024a)	–	60.42	68.42	64.17	26.30	22.95	43.56
Gemini-1.5 Flash (Team et al., 2024)	–	58.21	71.27	64.08	24.12	21.85	42.97
Claude-3 Haiku (Anthropic, 2024)	–	62.12	68.90	65.33	27.28	22.87	44.10
Open-source Multimodal Large Language Models							
Qwen2-VL (Wang et al., 2024)	72B	60.63	66.39	63.38	26.83	23.10	43.24
Intern-VL2 (Chen et al., 2024)	40B	56.63	65.15	60.59	24.77	21.16	40.88
Intern-VL2.5 (Chen et al., 2025b)	8B	55.61	65.41	60.11	23.12	21.91	41.62
LLaVA-v1.5 (Liu et al., 2024a)	7B	52.80	55.85	54.28	19.18	18.84	36.56
Qwen-VL-Chat (Bai et al., 2023)	7B	57.08	63.95	60.32	20.20	20.76	40.54
Qwen2.5-VL (Bai et al., 2025)	7B	61.29	63.57	62.41	25.81	21.99	42.20
Deepseek-VL-Chat (Lu et al., 2024)	7B	54.65	60.29	57.33	21.34	21.23	39.33
Phi3-Vision (Abdin et al., 2024)	7B	45.09	53.15	48.79	15.35	15.72	32.26
SciIE-related Models							
LLM-NERRE (Dagdelen et al., 2024)	7B	57.57	63.15	60.24	23.36	21.09	40.67
Eunomia (Ansari and Moosavi, 2024)	7B	65.44	68.92	67.14	28.72	25.86	46.50
BioWorkflow (Wang and Wang, 2025)	7B	61.90	63.49	62.68	24.91	22.31	42.50
Multimodal Agentic Framework (Ours)							
SudokuFill (Qwen2.5-VL)	7B	68.38	70.08	69.22	28.65	27.33	48.28
SudokuFill (Deepseek-VL-Chat)	7B	60.39	67.91	63.93	23.27	22.01	42.97
SudokuFill (GPT-5 Nano)	–	67.71	**78.84**	**72.85**	**34.38**	**30.80**	**51.83**

Table 1: **Main results on document-level adjuvant attribute extraction.** We report entity-level Precision/Recall/F1 and row-level performance. **Red:** best; Blue: second best.

Overall Score: we define the Overall metric as the arithmetic mean of Entity-level and Row-level F1.

Baselines We assess the effectiveness of **SudokuFill** on both open-source and closed-source MLLMs, and compared it against the following models: (1) **Closed-source MLLMs**, including GPT-4o, GPT-5 Nano, Gemini-1.5 Flash, and Claude-3 Haiku; (2) **Open-source MLLMs** across various scales (4B–72B), such as the Qwen2/2.5-VL series, Intern-VL2/2.5, DeepSeek-VL-Chat, Llava-v1.5 and Phi3-Vision; (3) **SciIE-related Models** including LLM-NERRE, Eunomia, and BioWorkflow, which are specifically designed for scientific domain extraction. For a fair comparison, all MLLM baselines are implemented using a sequential extraction strategy without priority scheduling. In this setting, models extract schema fields in a fixed order, without dynamic scheduling or multi-agent deliberation. SciIE-specialized models follow their original protocols adapted to our benchmark.

4.2 Experiment Results

Table 1 presents the performance of **SudokuFill** compared to a wide range of baselines. Our analysis reveals systematic patterns that validate the proposed framework.

A primary observation is that **SudokuFill** is effective across diverse backbones. Across both 7B-class open-source and frontier closed-source MLLMs, **SudokuFill** consistently improves performance. Specifically, compared to their vanilla versions, **SudokuFill** improves the Overall score by **6.08%** for Qwen2.5-VL, **3.64%** for DeepSeek-VL-Chat, and **8.27%** for GPT-5 Nano.

The most striking specific result is that **SudokuFill** with Qwen2.5-VL (7B) achieves an Overall score of 48.28%, surpassing the vanilla GPT-4o (47.72%). This result highlights that reformulating a massive long-context task into a sequence of localized, high-precision extraction rounds can reduce the reliance on model parameter scale. It further suggests that role division and iterative reuse are effective design choices for document-level SciIE.

A consistent pattern across all baselines is the pronounced drop from Entity-level to Row-level evaluation, reflecting a coherence gap in which a single field error can invalidate an entire record. Even SciIE-specialized models such as Eunomia are highly sensitive to this issue. In contrast,

Interaction Setting	Entity-F1	Row-F1	Overall
Reuse + Debate Rounds			
Reuse + 1-round debate	68.03 ± 0.62	27.16 ± 0.50	47.60
Reuse + 3-round debate	70.42 ± 0.55	28.48 ± 0.43	49.45
Reuse + adaptive-until-convergence debate	**72.34 ± 0.33**	**30.17 ± 0.24**	**51.26**
Debate Only (No Reuse)			
Adaptive debate without reuse	66.04 ± 0.49	23.33 ± 0.37	44.84

Table 2: **Interaction analysis of cross-query reuse and multi-agent debate under GPT-5 Nano.** Each setting is repeated three times. Best values are highlighted in red. Results show that reuse becomes substantially more effective and stable when paired with sufficiently strong multi-round debate.

SudokuFill demonstrates substantially stronger row-level extraction performance: the GPT-5 Nano variant achieves a Row-level accuracy of 34.38%, outperforming its vanilla counterpart by 8.08 percentage points. Importantly, we do not attribute this gain to cross-query reuse in isolation. Rather, the evidence suggests that the row-level improvement is associated with the interaction between reusable global constraints and sufficiently strong multi-agent deliberation. To clarify this mechanism, we provide an additional controlled comparison in Section 4.3.

4.3 Interaction between Reuse and Debate

To better understand the source of the row-level improvement, we conduct an additional controlled comparison focusing on the interaction between cross-query reuse and debate strength under the GPT-5 Nano backbone. As shown in Table 2, when reuse is enabled, increasing deliberation depth from one round to three rounds and then to adaptive-until-convergence yields a clear and stable improvement trend. Specifically, Row-F1 increases from 27.16 ± 0.50 to 28.48 ± 0.43 and further to 30.17 ± 0.24, while the Overall score rises from 47.60 to 49.45 and then to 51.26. The variance also decreases as debate becomes stronger, indicating that deeper deliberation not only improves extraction quality but also stabilizes the reuse process.

These results suggest that the benefit of the Global Filling State is not fully realized in isolation. Reuse provides structured anchors and cross-field constraints, but sufficiently strong multi-round debate is needed to validate, revise, and consolidate these anchors before they can reliably support downstream fields. This interpretation is further supported by the debate-only setting without reuse, which reaches only 44.84 Overall and 23.33 Row-F1, substantially below the reuse-enabled settings. Taken together, the evidence indicates that the row-level gains of **SudokuFill** are better explained by the coupling between reuse-based constraint propagation and debate-based error correction, rather than by either component alone.

4.4 Ablation Study

To investigate the contribution of each component in **SudokuFill**, we perform ablation studies with GPT-5 Nano as the backbone.

Variations Setup We design four categories of variants to isolate the impact of our core modules: **(1) w/o Scheduling** replaces the Stage I priority sequence with three random seeds (1-a, b, c) and a fixed schema order (1-d) to evaluate the "easy-to-hard" filling logic. **(2) w/o Cross-query Reuse (Ablation 2)** disables the document-level global filling state and the column-view agent $\mathcal{A}_{col}$, reverting to isolated field extraction. **(3) w/o Multi-agent Debate (Ablation 3)** simplifies Stage II to a single round of debate to assess the necessity of iterative deliberation. **(4) w/o Constraint Agents (Ablation 4)** systematically removes the row-view agent (4-a), the column-view agent (4-b), or both (4-c) to examine the synergy of structural constraints.

Results As shown in Table 3, the ablation results highlight several key insights. Removing priority scheduling (Ablation 1) causes only a modest decline, with the Overall score decreasing by about 1.5%. This suggests that cross-field reuse provides robustness to sub-optimal execution orders. In contrast, Ablation 3 leads to a larger drop, with the score falling to 44.46%, indicating that accurate field-level extraction is essential to the progressive framework. Furthermore, the substantial drop in Ablation 2 (to 47.39%) reinforces the necessity of the global filling state itself. These findings collectively align with our Sudoku-style hypothesis: while the order of filling (Stage I) provides an optimized path, the reliability of the information within each cell (Stage II) is the deciding factor for global convergence. Finally, Ablation 4 examines the role of constraint agents. Removing the $\mathcal{A}_{row}$ (4-a)

Ablation Setting	Entity-level			Row-level		Overall
	P	R	F1	Acc	F1	Avg F1
Full System						
Full: Stage I (scheduling) + Stage II (Extraction)	**67.71**	**78.84**	**72.85**	**34.38**	**30.80**	**51.83**
Ablation-1: w/o scheduling						
1-a Random order (seed=1)	66.58	75.46	70.74	32.14	30.03	50.39
1-b Random order (seed=2)	67.05	76.17	71.32	33.91	30.66	50.99
1-c Random order (seed=3)	66.52	73.78	69.96	31.27	29.12	49.54
1-d Schema fixed order	66.87	76.54	71.38	33.56	29.81	50.60
*Ablation-2: w/o cross-query reuse**						
2 Disable query-result reuse across fields	65.60	70.29	67.86	28.34	26.92	47.39
Ablation-3: w/o multi-agent debate						
3 Single-round regular debate	62.78	68.20	65.38	27.74	23.53	44.46
Ablation-4: w/o constraint agents						
4-a w/o $\mathcal{A}_{row}$	66.69	75.69	70.90	32.78	29.20	50.05
4-b w/o $\mathcal{A}_{col}$	67.38	77.55	72.11	33.84	30.71	51.41
4-c w/o both $\mathcal{A}_{row}$ and $\mathcal{A}_{col}$	65.50	73.21	69.14	31.94	28.75	48.95

Table 3: **Ablation study of SudokuFill under a fixed backbone (GPT-5 Nano).** *To disable cross-query reuse, we remove the document-level filling context built from previously converged queries; consequently, the column-view agent $\mathcal{A}_{col}$ is also disabled to avoid inadvertent access to resolved fields through the shared context.

or $\mathcal{A}_{col}$ (4-b) leads to a steady decline in performance, while their concurrent removal (4-c) yields the largest drop among these variants. This confirms that row-level and column-level constraints provide complementary structural signals.

4.5 Domain-aware Prompting vs. Rule-free Prompting

To quantify the impact of lightweight schema guidance, we evaluated SudokuFill under a rule-free setting, where only the list of schema fields is provided (without domain norms, compatibility cues, or schema-specific constraints), and agents rely purely on their own reasoning. Table 4 reports the results across three backbones on the AdjuvantIE benchmark.

As shown, removing domain-awareness leads to a consistent but moderate drop in performance across all backbones (on average 3–6 points in Overall), indicating that the framework remains practically usable even without domain-specific prompts. More importantly, even in this rule-free setting, SudokuFill often matches or exceeds much larger models or pipelines that rely on stronger domain rules, and rarely underperforms the corresponding vanilla backbone with domain prompting. These results further support the view that we are not claiming "no schema needed"; rather, SudokuFill is largely reusable as a general agentic framework, with lightweight schema guidance primarily serving to unlock its best performance rather than being a strict requirement for effectiveness. This highlights the framework's practicality and transferability.

5 Further Analysis

We further analyze the experimental results, focusing on two findings: test-time scaling in our multi-agent system (Section 5.1) and the selection of the confidence threshold in Stage I (Section 5.2).

5.1 Test-time Scaling

We analyze test-time scaling by relating performance to the document-level budget (tokens and agent calls), primarily controlled by the number of Stage II deliberation rounds. Fig 3 shows test-time scaling across all backbones: Overall F1 increases monotonically as budget rises from 160k to 1200k tokens. This trajectory indicates that document-level SciIE is a compute-intensive reasoning task rather than a static retrieval problem. With additional thinking time, the multi-agent system better leverages increased inference compute to resolve layout ambiguities and refine evidence localization through the global filling state.

The results further reveal a significant scale compensation effect enabled by our iterative architecture. While GPT-5 Nano exhibits the highest scaling efficiency, climbing from 43.84% to 51.83% as it absorbs more budget, the 7B-class Qwen2.5-VL manages to reach a performance plateau of 48.28%

Model	Domain-aware	Performance		
		Entity-F1	Row-F1	Overall
Domain-aware prompting				
SudokuFill (Qwen2.5-VL)	✓	69.22	27.33	48.28
SudokuFill (GPT-5 Nano)	✓	**72.85**	**30.80**	**51.83**
SudokuFill (DeepSeek-VL-Chat)	✓	63.93	22.01	42.97
Rule-free prompting				
SudokuFill (Qwen2.5-VL)	✗	63.48	21.54	42.51
SudokuFill (GPT-5 Nano)	✗	67.37	24.97	46.17
SudokuFill (DeepSeek-VL-Chat)	✗	60.21	19.88	40.15

Table 4: **Domain-aware prompting vs. rule-free prompting on the AdjuvantIE benchmark.** ✓indicates the presence of lightweight schema guidance, while ✗indicates the rule-free setting. The results show that even without domain-specific prompts, SudokuFill remains competitive across different backbones.

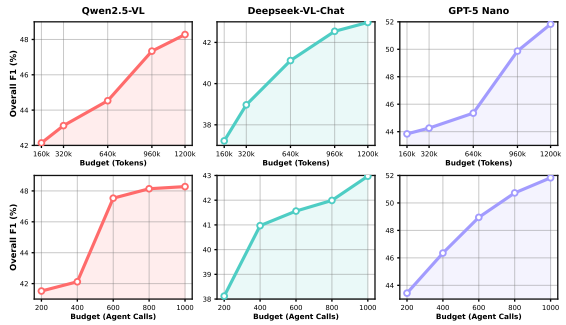

Figure 3: Test-time scaling under three backbones.

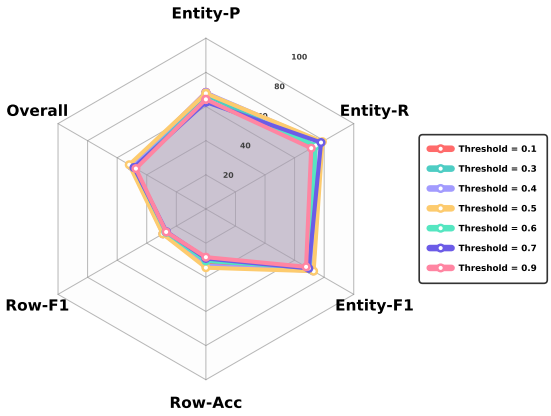

Figure 4: Radar plot for confidence-threshold selection.

at its higher tiers, notably surpassing the low-budget performance of the frontier GPT-5 Nano model. This gap narrowing suggests that structured, recursive reasoning can partially offset limited parameter scale. SudokuFill effectively converts additional inference-time compute into improved deliberative consistency, enabling smaller open-source models to approach strong proprietary baselines.

5.2 Stage I Confidence Threshold Selection

We evaluate the sensitivity of the confidence threshold τ in Stage I by varying it from 0.1 to 0.9 (SudokuFill - GPT-5 Nano), observing that performance follows a stable trajectory that peaks at $\tau = 0.5$ (51.83% Overall, 34.38% Row-Acc), as shown in Fig 4. This optimal point is intuitive as it serves as the natural probabilistic decision boundary, effectively balancing the inclusion of likely page-level evidence with the exclusion of low-confidence noise. Below this threshold, the system exhibits a moderate decline; while signal recall is high, the global filling state becomes contaminated with noisy anchors, triggering minor cascading errors during the multi-agent deliberation phase. In contrast, a slight decline occurs as τ exceeds 0.5, with the Overall score adjusting to 49.62% at $\tau = 0.7$ and 48.93% at $\tau = 0.9$. This steady performance where Row-Acc remains above 29%, indicates that Stage II is resilient, successfully utilizing even a limited set of seeds to populate knowledge base. These findings confirm that $\tau = 0.5$ represents the ideal calibration point for stabilizing document-level IE while maintaining high tolerance for threshold variations.

6 Conclusion

We proposed SudokuFill, a multi-agent framework that reframes document-level SciIE as a progressive, constraint-driven reasoning process. By iteratively reusing high-confidence extractions as structured constraints, SudokuFill emphasizes global consistency over isolated field prediction. Experiments on a newly constructed document-level vaccine adjuvant benchmark demonstrate that this progressive reuse paradigm consistently improves record-level coherence across model backbones, enabling smaller models to rival or surpass larger MLLMs. These results suggest that structured iterative reasoning is a more effective lever than parameter scaling for long-document SciIE.

Limitations

SudokuFill incurs higher inference time overhead due to multi-round agent interaction and iterative reuse, which may limit deployment under tight latency and budget constraints. Progressive reuse also risks error propagation, as early mistakes may affect subsequent extractions. Future work will explore more selective scheduling, earlier stopping, and confidence checks to mitigate these issues.

Acknowledgements

This work was supported by the National Key R&D Program of China (No. 2023YFF0725600), Lingang Laboratory (Grant No. LGL-2616-04), and Open Funding Project of the State Key Laboratory of Biopharmaceutical Preparation and Delivery (No. 2024KF-06).

References

Marah Abdin, Jyoti Aneja, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

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

Mehrad Gholizadeh Ansari and Seyed Mohamad Moosavi. 2024. Agent-based learning of materials datasets from scientific literature. *Digital Discovery*.

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jacob Berlin and Amihai Motro. 2002. Database schema matching using machine learning with feature selection. In *International Conference on Advanced Information Systems Engineering*, pages 452–466. Springer.

Honghao Chen, Hongxuan Liu, Yishen Tew, Xiaotian Ren, Xiaojin Tang, and Xiaonan Wang. 2025a. Distilling knowledge from catalysis literature with long-context large language model agents. *ACS Catalysis*, 15(21):18244–18254.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. 2025b. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *https://arxiv. org/abs/2412.05271*.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. 2024. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *Science China Information Sciences*, 67(12):220101.

Dimitrios Christofidellis, Giorgio Giannone, Jannis Born, Ole Winther, Teodoro Laino, and Matteo Manica. 2023. Unifying molecular and textual representations via multi-task language modelling. In *International Conference on Machine Learning*, pages 6140–6157. PMLR.

John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S Rosen, Gerbrand Ceder, Kristin A Persson, and Anubhav Jain. 2024. Structured information extraction from scientific text with large language models. *Nature communications*, 15(1):1418.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Mingliang Dou, Jijun Tang, Prayag Tiwari, Yijie Ding, and Fei Guo. 2024. Drug–drug interaction relation extraction based on deep learning: a review. *ACM Computing Surveys*, 56(6):1–33.

Luis Espinosa-Anke and Horacio Saggion. 2014. Applying dependency relations to definition extraction. In *International Conference on Applications of Natural Language to Data Bases/Information Systems*, pages 63–74. Springer.

Luke PJ Gilligan, Matteo Cobelli, Valentin Taufour, and Stefano Sanvito. 2023. A rule-free workflow for the automated generation of databases from scientific literature. *npj Computational Materials*, 9(1):222.

Cyril Goutte and Eric Gaussier. 2005. A probabilistic interpretation of precision, recall and f-score, with

implication for evaluation. In *European conference on information retrieval*, pages 345–359. Springer.

Tanishq Gupta, Mohd Zaki, NM Anoop Krishnan, and Mausam. 2022. Matscibert: A materials domain language model for text mining and information extraction. *npj Computational Materials*, 8(1):102.

Monika Jain, Kuldeep Singh, and Raghava Mutharaju. 2023. Reonto: A neuro-symbolic approach for biomedical relation extraction. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 230–247. Springer.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Yang Li, Mengting Zhang, Zhixiong Zhang, and Yajiao Wang. 2024. Decoding the essence of scientific knowledge entity extraction: An innovative mrc framework with semantic contrastive learning and boundary perception. In *Proceedings of the 24th ACM/IEEE Joint Conference on Digital Libraries*, pages 1–12.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024b. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Sijia Liu, Feichen Shen, Vipin Chaudhary, and Hongfang Liu. 2017. Mayonlp at semeval 2017 task 10: Word embedding distance pattern for keyphrase classification in scientific publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 956–960.

Xiong Liu, Greg L Hersch, Iya Khalil, and Murthy Devarakonda. 2021. Clinical trial information extraction with bert. In *2021 IEEE 9th International Conference on Healthcare Informatics (ICHI)*, pages 505–506. IEEE.

Micha Livne, Zulfat Miftahutdinov, Elena Tutubalina, Maksim Kuznetsov, Daniil Polykovskiy, Annika Brundyn, Aastha Jhunjhunwala, Anthony Costa, Alex Aliper, Alán Aspuru-Guzik, et al. 2024. nach0: multimodal natural and chemical languages foundation model. *Chemical Science*, 15(22):8380–8389.

Zhang Liwei. 2022. Chinese technical terminology extraction based on dc-value and information entropy. *Scientific Reports*, 12(1):20044.

Ivan Lopez, Akshay Swaminathan, Karthik Vedula, Sanjana Narayanan, Fateme Nateghi Haredasht, Stephen P Ma, April S Liang, Steven Tate, Manoj Maddali, Robert Joseph Gallo, et al. 2025. Clinical entity augmented retrieval for clinical information extraction. *npj Digital Medicine*, 8(1):45.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, et al. 2024. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*.

OpenAI. 2024a. Introducing gpt-5. Accessed: 2025-08-07.

OpenAI. 2024b. GPT4o-mini. Accessed: 2025-05-01.

Martín Pérez-Pérez, Tânia Ferreira, Gilberto Igrejas, and Florentino Fdez-Riverola. 2022. A deep learning relation extraction approach to support a biomedical semi-automatic curation task: the case of the gluten bibliome. *Expert Systems with Applications*, 195:116616.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

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

Pranav Shetty, Arunkumar Chitteth Rajan, Chris Kuenneth, Sonakshi Gupta, Lakshmi Prerana Panchumarti, Lauren Holm, Chao Zhang, and Rampi Ramprasad. 2023. A general-purpose material property data extraction pipeline from large polymer corpora using natural language processing. *npj Computational Materials*, 9(1):52.

Manmohan Singh and Derek O'Hagan. 1999. Advances in vaccine adjuvants. *Nature biotechnology*, 17(11):1075–1081.

Angelika Storrer and Sandra Wellinghoff. 2006. Automated detection and annotation of term definitions in german text corpora. In *LREC*, pages 2373–2376.

Yuqi Sun, Weimin Tan, Zhuoyao Gu, Ruian He, Siyuan Chen, Miao Pang, and Bo Yan. 2025. A data-efficient strategy for building high-performing medical foundation models. *Nature Biomedical Engineering*, pages 1–13.

Matthew C Swain and Jacqueline M Cole. 2016. Chemdataextractor: a toolkit for automated extraction of chemical information from the scientific literature. *Journal of chemical information and modeling*, 56(10):1894–1904.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Dan Tian, Mingchao Li, Yang Shen, and Shuai Han. 2023. Intelligent mining of safety hazard information from construction documents using semantic similarity and information entropy. *Engineering Applications of Artificial Intelligence*, 119:105742.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *Preprint*, arXiv:2409.12191.

Yidan Wang and Jiayin Wang. 2025. Bioworkflow: Retrieving comprehensive bioinformatics workflows from publications. *Briefings in Bioinformatics*, 26(6):bbaf571.

Rui Zhang, Jiawang Zhang, Qiaochuan Chen, Bing Wang, Yi Liu, Quan Qian, Deng Pan, Jinhua Xia, Yinggang Wang, and Yuexing Han. 2023. A literature-mining method of integrating text and table extraction for materials science publications. *Computational Materials Science*, 230:112441.

Wei Zhang, Qinggong Wang, Xiangtai Kong, Jiacheng Xiong, Shengkun Ni, Duanhua Cao, Buying Niu, Mingan Chen, Yameng Li, Runze Zhang, et al. 2024. Fine-tuning large language models for chemical text mining. *Chemical science*, 15(27):10600–10611.

Zhiling Zheng, Oufan Zhang, Christian Borgs, Jennifer T Chayes, and Omar M Yaghi. 2023. Chatgpt chemistry assistant for text mining and the prediction of mof synthesis. *Journal of the American Chemical Society*, 145(32):18048–18062.

Yangqiaoyu Zhou, Haokun Liu, Tejes Srivastava, Hongyuan Mei, and Chenhao Tan. 2024. Hypothesis generation with large language models. *arXiv preprint arXiv:2404.04326*.

Appendix

A More Details of Stage I

Stage I serves as the "probing and scheduling" phase of **SudokuFill**, aimed at identifying high-confidence signals and establishing an optimal execution order to minimize information loss. This appendix details the semantic normalization, the confidence calibration rubric, and the ranking logic.

A.1 Field-to-Query Instantiation

To standardize extraction targets across heterogeneous agents and enable cross-query reuse, we convert each schema field $f \in \mathcal{F}$ into a field-grounded query unit $q = \langle f, \phi(f) \rangle$. Here $\phi(f)$ is a templated semantic specification that defines what to extract, the expected value format, optional normalization rules, and a strict evidence requirement with verifiable localization. All StageI agents are constrained to output a unified triple (V, C, E), where V is the proposed value, $C \in [0, 1]$ is the agent's internal confidence estimate, and E records evidence localization. Crucially, $\phi(f)$ also includes a *reusable context slot* that accommodates resolved query results from earlier rounds: once anchor fields are prioritized and converged into the global filling state, their confirmed outputs are re-injected into subsequent queries in the same query-normalized form, so later extraction is conditioned on standardized constraints rather than ad-hoc textual concatenation. This query abstraction makes field semantics explicit to the agents and enables stable cross-round and cross-query information flow, shrinking the search space and reducing ambiguity for downstream fields. In our benchmark, we instantiate ten core fields into queries following the same template: *Adjuvant_Name, Category, Sub_type, Composition, Morphology, Particle_Size, Particle_Structure, Target, Target_Cell,* and *Combination_mode*. Full prompt instances for $\phi(f)$ are provided in Table 5.

A.2 Confidence Calibration Rubric

Each page agent outputs an internal confidence score $C \in [0, 1]$ together with (V, E). We emphasize that confidence is not calibrated across heterogeneous agents and is therefore not used as an absolute, cross-agent comparable quantity; instead, StageI uses C only as a within-agent reliability signal to (i) compute query-level summaries for scheduling and (ii) define whether a page provides an effective response for participation statistics. To make C interpretable and consistent across queries, we adopt a rubric aligned with evidence strength and verifiability (details in Table 6.). High confidence is reserved for cases where the proposed value is explicitly stated and uniquely supported by a localized snippet (e.g., a table cell, a clear sentence, or an unambiguous caption). Medium confidence corresponds to grounded but slightly under-specified cases that require light normalization or minor aggregation, while low confidence reflects weak or indirect cues that are not sufficiently reliable for scheduling decisions. Following this rubric, we define an effective participation by $C > 0.5$ when computing $\mathrm{Part}(q)$, which filters noisy responses triggered by weak matches and better reflects whether a query exhibits concentrated, actionable signals across pages. We provide an empirical threshold sweep and the rationale for choosing 0.5 in Section 5.2.

A.3 Rank Agent: Multi-Factor Priority Scheduling

Stage I constructs a page–query matrix $\mathcal{M}$ where each entry $\mathcal{M}[p, q] = (V_{p,q}, C_{p,q}, E_{p,q})$ records a page agent's candidate for query q on page p. The rank agent $\mathcal{A}_R$ takes $\mathcal{M}$ as its sole runtime input; all additional signals are derived summaries provided in the prompt to improve decision transparency. Concretely, for each query q, we compute $\mathrm{MaxConf}(q) = \max_p C_{p,q}$ as a proxy for the strongest identifiable signal within the document, and $\mathrm{Part}(q) = |\{p \mid C_{p,q} > 0.5\}|$ as a proxy for signal concentration and robustness across pages. In addition, the prompt provides a schema-implied dependency specification $\textsc{FieldDependency}(\mathcal{Q})$, which encodes precedence constraints among fields so that anchor fields that are likely to constrain others can be scheduled earlier. The rank agent outputs a prioritized query sequence π by jointly considering these factors, with $\mathrm{MaxConf}(q)$ as the primary signal, $\mathrm{Part}(q)$ as a stability cue, and $\textsc{FieldDependency}(\mathcal{Q})$ as a soft constraint to encourage an easy-to-hard execution order that maximizes downstream reuse.

A.4 System Prompts for Stage I

A.4.1 Page Agent Probing Prompt

The page agent prompt instructs the model to (i) scan only the assigned page for a given query q, (ii) propose a candidate value $V_{p,q}$ only when grounded evidence is present, (iii) report an internal confidence score $C_{p,q}$ according to the

Field f	Query input: definition + resolved context
Adjuvant_Name	**Field definition:** the name(s) of vaccine adjuvant(s) studied in the paper.
Resolved context: $\mathcal{G}^{(t)}$ (previously converged fields, if any).	
Output: (V, C, E) with verifiable localization.	
Category	**Field definition:** the adjuvant category explicitly supported by evidence {e.g., `particle`, `emulsion`, `molecular`, `inorganic_salt`, `other`, `composite`, `NA`}.
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	
Sub_type	**Field definition:** if `Category` is `particle` or `molecular`, classify using subtype sets; otherwise `NA`.
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	
Composition	**Field definition:** merge all formulation descriptions into a unified component list grounded in the paper.
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	
Morphology	**Field definition:** extract explicitly stated morphology/form (e.g., hydrogel, microneedle, solution); otherwise `NA`.
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	
Particle_Size	**Field definition:** particle size description (prefer numeric value + unit if available).
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	
Particle_Structure	**Field definition:** structural descriptors (e.g., core–shell, porous, multilamellar) if explicitly evidenced; otherwise `NA`.
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	
Target	**Field definition:** for `Category=molecular`, immune target/pathway explicitly stated; otherwise `NA`.
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	
Target_Cell	**Field definition:** for `Category=molecular`, target cell type(s) explicitly mentioned; otherwise `NA`.
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	
Combination_mode	**Field definition:** for `Category=composite`, combination mode from {loading, chemical_conjugation, linker_conjugation, mixing, fusion_expression, other, NA}; otherwise `NA`.
Resolved context: $\mathcal{G}^{(t)}$.	
Output: (V, C, E) with verifiable localization.	

Table 5: A field-to-query interface. Each agent receives the field definition and the resolved filling context $\mathcal{G}^{(t)}$ from previously converged queries, and outputs (V, C, E) with verifiable evidence localization.

rubric in Appendix A.2, and (iv) return verifiable evidence localization $E_{p,q}$ to support auditing. The output is constrained to the structured triple $(V_{p,q}, C_{p,q}, E_{p,q})$, which is used to populate $\mathcal{M}$ and to warm-start Stage II (details in Table 12).

A.4.2 Rank Agent Scheduling Prompt

The rank agent prompt consumes the serialized page–query matrix $\mathcal{M}$ and receives prompt-only summaries including $\{\mathrm{MaxConf}(q)\}_{q\in\mathcal{Q}}$, $\{\mathrm{Part}(q)\}_{q\in\mathcal{Q}}$, and $\textsc{FieldDependency}(\mathcal{Q})$. It is instructed to output a total order π over queries, optionally accompanied by brief rationales that reference these signals. This prompt ensures that Stage I produces a deterministic, reproducible scheduling policy while keeping the only runtime evidence source as the page-level probing outcomes in $\mathcal{M}$ (details in Table 13).

A.5 Stage I Algorithm Pseudocode

Finally, we present the Stage I process in pseudocode below (details in algorithm 1).

B More Details of Stage II

Multi-agent deliberation can over-commit to an early majority when evidence is sparse or unevenly distributed across pages, so that minority but critical objections are not sufficiently surfaced and verified. To improve robustness, the coordinator $\mathcal{A}_C$ does not follow a fixed turn-taking routine. Instead, it maintains an explicit round history $H^{(t)}(q)$ and adaptively selects among three deliberation modes conditioned on this history. Concretely, $H^{(t)}(q)$ compactly records the current leading candidate, the main supporting and opposing evidence pointers, which objections remain unresolved, and each agent's within-agent confidence revisions across rounds. This makes the debate explicitly stateful and prevents later rounds from restarting from scratch.

Round-level execution (one round in detail). At round t for query q, the coordinator first assembles a shared round context that includes: the query specification $q = \langle f, \phi(f)\rangle$; the current

C	Confidence calibration rubric (discrete levels)
0.0	**No response / not applicable.** The query is irrelevant to this page or no candidate can be proposed; output `N/A` and note "no mention on this page" in E.
0.1	**Weak cue only.** Some related terms appear but the target value is not stated; evidence cannot support a concrete candidate (avoid proposing a value).
0.3	**Ambiguous candidate.** A citable location suggests a plausible candidate, but the mention is implicit or non-unique and requires substantial inference; keep below participation threshold.
0.5	**Minimally grounded (participation boundary).** A plausible candidate is supported by verifiable localization, but it may be incomplete, non-unique, or require light normalization; evidence exists but is not decisive.
0.7	**Explicit but needs disambiguation.** The value is explicitly stated with verifiable evidence, yet multiple candidates are present or selection depends on additional context (other pages or resolved fields).
0.9	**Explicit, unique, and well-supported.** The value is clearly and uniquely supported on this page, typically corroborated by multiple cues (e.g., table + caption, repeated mentions), with little room for alternatives.
1.0	**Decisive evidence.** The value is unambiguous, uniquely specified in a definitive form (e.g., exact table cell with headers or a clear statement) and fully consistent with all available cues.

Table 6: **Confidence calibration rubric for Stage I Page Agents.** $C \in [0, 1]$ is an internal reliability estimate aligned with evidence verifiability and value determinacy. We count effective participation as $C > 0.5$ when computing $\mathrm{Part}(q)$.

candidate pool $\mathcal{C}^{(t)}(q)$ (deduplicated values with accumulated evidence pointers); and a compact snapshot $H^{(t-1)}(q)$ (the previous leading candidate and open objections). The round then follows a stable three-phase routine. In Phase 1 (position statement), all agents independently publish their current stance with evidence: each page agent outputs a possibly revised (V, C, E) grounded on its assigned page, while each row-/column-view constraint agent outputs (V^{sup}, C, E) where V^{sup} must be selected from existing candidates and E cites retrieved constraints or cross-field consistency checks from the current global filling state $\mathcal{G}$. In Phase 2 (open rebuttal), agents directly respond to others by challenging evidence alignment (wrong entity, wrong condition, wrong table row/column), defending a candidate with additional on-page evidence, or revising their stance when an objection holds. In Phase 3 (consolidation), the coordinator merges equivalent values, attaches newly surfaced evidence to the corresponding candidate entry, records per-candidate support and opposition, and updates $H^{(t)}(q)$ with what changed in this round (which objections were answered, which remain open, and how each agent revised its own confidence relative to its previous stance). Finally, $\mathcal{A}_C$ decides the mode for the next round based on the updated history and evidence distribution.

Regular debate (default). Regular debate is used by default to aggregate multimodal evidence across pages and refine candidates through repeated cross-checking between page agents and constraint agents. In this mode, all agents participate symmetrically under the above three-phase routine. Page agents are encouraged to (i) introduce candidates only when they can provide verifiable localization, (ii) correct earlier misreads (unit mismatch, time-point mismatch, entity mismatch), and (iii) explicitly update their confidence to reflect whether new rebuttals strengthened or weakened their stance. Constraint agents act as structured "critics" that stress-test candidates against $\mathcal{G}$: they may endorse a candidate when it matches canonical patterns or co-occurrence regularities, or challenge it when it violates field dependencies or record-level consistency implied by previously converged fields. A regular round typically reduces ambiguity by converting free-form disagreement into checkable disputes about evidence alignment, and by shrinking the candidate pool to a small set of well-supported alternatives with explicit unresolved objections recorded in $H^{(t)}(q)$.

Anti-bias debate. Anti-bias debate is designed to mitigate the "echo chamber" failure mode in which a dominant early proposal is repeatedly reinforced while a minority but evidence-backed objection is not examined to the same standard. The coordinator activates this mode when the history indicates warning patterns such as: rapid group alignment in very few rounds without substantive examination of alternatives, and/or a persistent, evidence-backed objection that remains unaddressed while other agents continue to repeat endorsements. An anti-bias round keeps the same interfaces but changes

Algorithm 1 Stage I: Field Priority Scheduling

Require: Document $\mathcal{D}$, schema fields $\mathcal{F}$, Page Agents $\mathcal{A}_P$, Round Controller $\mathcal{A}_R$

$\mathcal{Q} \leftarrow \{\, q = \langle f, \phi(f) \rangle \mid f \in \mathcal{F} \,\}$ ▷ Field-to-query instantiation

for all $p \in \textsc{Pages}(\mathcal{D})$ **in parallel do**

  for all $q \in \mathcal{Q}$ **do**

    $(V_{p,q},\, C_{p,q},\, E_{p,q}) \leftarrow \mathcal{A}_P(p, q)$ ▷ MLLM-based agents for page probing

    $\mathcal{M}[p, q] \leftarrow (V_{p,q},\, C_{p,q},\, E_{p,q})$

  end for

end for

for all $q \in \mathcal{Q}$ **do**

  $\mathrm{MaxConf}(q) \leftarrow \max_p C_{p,q}$

  $\mathrm{Part}(q) \leftarrow |\{p \mid C_{p,q} > 0.5\}|$

end for

$\mathbf{P} \leftarrow \Big(\{\mathrm{MaxConf}(q)\}_{q\in\mathcal{Q}},\ \{\mathrm{Part}(q)\}_{q\in\mathcal{Q}},\ \textsc{FieldDependency}(\mathcal{Q})\Big)$ ▷ Prompt-only signals; template in Appendix

$\pi \leftarrow \mathcal{A}_R\big(\mathcal{M}\,;\,\mathbf{P}\big)$

$\mathcal{C}_0 \leftarrow \{(q, V_{p,q}, C_{p,q}, E_{p,q}) \mid q \in \mathcal{Q}, p \in \textsc{Pages}(\mathcal{D})\}$ ▷ Warm-start candidates

return $\pi,\ \mathcal{C}_0$

the speaking priority and the tasks. First, the coordinator explicitly grants the "microphone" to the minority agents and requests a checkable objection: what exact evidence contradicts the current dominant value, and which alternative value (selected from the existing pool) the objection supports. Second, the coordinator converts this objection into an explicit verification task for the previously supporting agents, asking them to re-check their own pages or constraints specifically for (i) counter-evidence that refutes the objection, (ii) missing qualifiers that reconcile the disagreement, or (iii) overlooked evidence that supports the alternative. Third, the coordinator forces a focused response in $H^{(t)}(q)$: supporters must state whether and why they keep or revise their stance, and objectors must state whether the responses resolve the concern. The mode exits once the objection has been explicitly answered (e.g., the objector's confidence drops or the objection is shown inapplicable); otherwise, if the objection is validated, the dominant candidate is weakened and the system returns to regular debate with an updated candidate landscape rather than continuing to reinforce the old majority.

Re-thinking. Re-thinking addresses the one-off signal risk: when the leading candidate is supported by only a single page source or effectively by a single agent, the decision becomes vulnerable to spurious matches and local hallucinations. The coordinator triggers re-thinking when the leading evidence is narrowly concentrated (e.g., only one page yields $C > 0.5$ or only one agent consistently produces the candidate with non-trivial confidence). In a re-thinking round, the coordinator treats the current leading value as a hypothesis and actively queries previously non-participating or low-confidence page agents with a targeted follow-up: under this hypothesis, search your assigned page for corroboration, contradiction, or an alternative mention, and return verifiable localization if found. The coordinator then updates the candidate pool and history based on outcomes: if corroboration emerges from multiple pages, the hypothesis is promoted with aggregated multi-page evidence; if a strong contradiction emerges, the system returns to regular debate to re-evaluate competing candidates under the new evidence; if all queried agents report no signal, the coordinator keeps the hypothesis as the best available single-source answer but records this limitation explicitly in $H^{(t)}(q)$ and maintains an appropriately conservative confidence.

Convergence criteria. Because confidence scores are not directly comparable across heterogeneous agents, $\mathcal{A}_C$ judges convergence by temporal trends rather than absolute confidence values. Specifically, it monitors (i) whether the leading candidate remains unchanged across successive rounds, (ii) whether each agent's within-agent confidence updates become progressively smaller (from large revisions to minor adjustments), and (iii) whether any evidence-backed objection remains unresolved in $H^{(t)}(q)$. Intuitively, this produces an "annealing-style" stabilization: early rounds allow large confidence and stance revisions to encourage exploration and correction, while later rounds naturally transition to small refinements once evidence has been exhausted. The coordinator declares convergence when the leading candidate is stable for multiple rounds, most agents only make marginal within-agent confidence updates, and no unresolved strong objection persists. Upon convergence, $\mathcal{A}_C$ outputs the final value V with an aggregated evidence set E (spans/cells/captions across pages when available), and writes (q, V, E) back into the

global filling state $\mathcal{G}$ so that subsequent queries can reuse the resolved field as standardized context and constraint.

B.1 System Prompts for Stage II

This section presents the core prompts of Stage II, including the Page Agent Debate Prompt, the Constraint Row Agent Prompt, the Constraint Column Agent Prompt, and the Coordinator Prompt; these form the backbone of the multi-agent deliberation mechanism, guiding agents to evaluate, constrain, and coordinate candidate values in a consistent and verifiable manner (details in Table 14, 15, 16, 17).

C Dataset

C.1 Data Collection and Selection Criteria

The benchmark was constructed to represent the frontier of vaccine adjuvant research. We performed a targeted literature search using the query: TOPIC=(("vaccine*" OR "adjuvant*" OR "immuniz*") AND ("nanoparticle*" OR "particle*" OR "microcapsule*" OR "capsule*")). To ensure high academic impact and data quality, we restricted the source journals to top-tier publications (CNS Q1 Top), including:

- Multidisciplinary / Nature-Science Family: Nature, Science, Cell, Nature Medicine, Nature Biotechnology, Nature Materials, Science Immunology, Science Advances, etc.
- Specialized Nanotechnology Biomaterials: Biomaterials, Advanced Materials (AM), Advanced Functional Materials (AFM), ACS Nano, Nano Letters, Journal of Controlled Release (JCR), Small, Nano-Micro Letters, etc.

From an initial pool of approximately 1,200 papers (2017–2026), we performed manual expert filtering to select the 250 most relevant papers containing end-to-end experimental data.

Data Access and Consent The data used in this study was obtained through a formal application process to ensure ethical compliance and proper usage. For inquiries regarding data access, consent protocols, or to request permission for research purposes, please contact the data management team directly at liyang2022@mail.las.ac.cn. We ensure that all data distribution is contingent upon the recipient's agreement to our privacy and usage terms. Ethics Committee Approval Yes, the data collection and usage protocol for this study were formally approved by the Institutional Review Board (IRB) / Ethics Committee of National Science Library, Chinese Academy of Sciences (LAS). All procedures were conducted in strict accordance with the approved guidelines to ensure the protection of participants' privacy and data security.

C.2 Five-Step Expert Annotation Protocol

To capture the complex logic of scientific discovery, our annotation process followed a rigorous five-step heuristic protocol: 1. Lead Object Identification: Identify the primary research objects by analyzing the frequency of experimental groups in comparative figures (e.g., the most frequent contrast pair like TLR7-alum vs. TLR7-NP). 2. Contextual Definition: Locate the first mention of identified names to establish a preliminary definition of the adjuvant system. 3. Novelty Verification: Determine if the group contains a novel adjuvant or an established delivery system. 4.Category Classification: Classify the adjuvant into categories (Particle, Molecular, Inorganic Salt, Composite, or Other) based on semantic cues like "size/SEM" for particles or "receptor/molecular formula" for molecular types. 5.Attribute Extraction: Fill the schema (e.g., Composition, Morphology, Particle_Size) by localizing specific characterization snippets in Results or Methods.

C.3 Quality Assurance and Statistics

The reliability of scientific data extraction depends heavily on domain-specific knowledge. Our annotation team consisted of two primary annotators, both of whom are PhD candidates specializing in vaccine adjuvants and biomaterials. This ensured a deep understanding of complex chemical nomenclatures, immunological mechanisms, and experimental methodologies. A senior scientist with over 10 years of experience in adjuvant research served as the final arbitrator to resolve discrepancies and ensure the highest level of ground-truth accuracy.

We implemented a rigorous three-phase workflow to eliminate subjective bias and ensure data consistency:

Phase 1: Double-Blind Independent Labeling. The two primary annotators independently extracted records from the raw PDFs following the five-step protocol (Sec. C.2). They were blinded to each other's results to prevent cross-influence.

Phase 2: Consistency Assessment. We conducted a systematic consistency check using Co-

hen's Kappa for categorical fields (e.g., Category, Sub_type) and F1-score for free-text fields (e.g., Composition, Morphology). The inter-annotator agreement reached an initial high threshold (Kappa > 0.82), indicating the protocol's clarity.

Phase 3: Expert Arbitration. Discrepancies identified in Phase 2, often involving ambiguous experimental groups or implicit evidence, were submitted to the senior arbitrator. The arbitrator reviewed the raw evidence in the PDF to make a final decision, resulting in the finalized "Gold Standard" dataset.

C.4 Complexity Analysis: The "Needle in a Haystack" Challenge

The benchmark presents a unique challenge for AI: the average paper length is 20 pages, yet the key evidence for a single adjuvant attribute is often buried in a single sentence or a sub-figure caption. Moreover, the multi-object nature (avg. 4.3 adjuvants/-paper) requires the model to maintain long-range spatial awareness to avoid cross-contamination between experimental groups. This "Sudoku-like" dependency, where knowing the Category helps constrain the Particle_Structure, validates the necessity of our Progressive Filling framework over one-pass extraction.

C.5 Comparison with Existing Scientific IE Datasets

First, existing scientific IE datasets can roughly be divided into two types: sentence-, paragraph-, or abstract-level IE, where the model processes relatively short text (MaterialIE1/2/3, DopingIE, Inpatient-recordIE), and document-level IE, where the model must search over entire papers (MOFIE2, BPMIE, CFSIE, UVIE), which is inherently a "needle-in-a-haystack" setting.

Our benchmark belongs to the second, harder category. All 250 documents are full-length CNS papers (average 20 pages). Each paper typically contains multiple adjuvants, and we exhaustively annotate all of them, resulting in over 10,000 entity-level instances and more than 1,000 row-level records, each with 10 fields. The dataset was constructed over six months by six domain experts (PhD level or above) in vaccine adjuvants. In terms of effective supervision (documents × adjuvants per document × fields per record), this dataset is already comparable to, or larger than, most prior document-level SciIE benchmarks. We summarize these statistics in Table 7.

C.6 Dataset Example

To illustrate the structure of the AdjuvantIE benchmark, we provide several representative examples of annotated records. (details in Table 8):

D More Details of Experiments

D.1 Evaluation on MaterialIE2

To further validate the generality of SudokuFill beyond our own benchmark, we evaluated it on the MaterialIE2 dataset. Table 9 presents entity-level F1 scores across multiple fields and the overall macro-F1 for two SciIE state-of-the-art models, two mainstream large language models, and SudokuFill-Qwen2.5-VL 7B. SudokuFill achieves the highest overall macro-F1, demonstrating its effectiveness on an independent scientific IE dataset.

D.2 Evaluation on Weak Baselines

To further validate the generality of SudokuFill across different model capacities, we evaluated several relatively weaker multimodal backbones, including LLaVA-v1.5, Phi3-Vision, and InternVL2.5. Table 10 summarizes the results on the AdjuvantIE benchmark. Under the same backbone configurations, SudokuFill consistently improves both entity-level and overall metrics over the vanilla counterparts. Although these weaker models have lower absolute performance than GPT-5 series models, the improvement trend remains consistent: structured reasoning and cross-field coordination systematically benefit models with limited intrinsic capacity.

D.3 Ordering Stability Analysis

To further characterize the robustness of SudokuFill, we analyzed the stability of the query ordering on 50 papers, each executed three times under identical backbone, prompts, and decoding settings. We found that the ordering is highly stable at the per-document level: 39 papers (78%) were identical across runs, while the remaining 11 papers (22%) exhibited at least one differing run. Variations across documents reflect the design, as the scheduler relies on document-specific textual evidence and confidence signals rather than fixed heuristics. Examining the 11 "stress set" papers, we observed that final extraction performance varies only marginally across runs, whereas the filling-time cost (number of steps/rounds until convergence) shows somewhat greater sensitivity, as summarized in Table 11. Differences in ordering

Dataset	Task Level	Test Samples	Number of Fields
MaterialIE1	Sentence	–	2
DopingIE	Sentence	77	4
MOFIE1	Abstract	255	5
MaterialIE2	Abstract	320	6
Inpatient-recordIE	Paragraph	500	9
MaterialIE3	Paragraph/Sentence	156 / 1330	5
MOFIE2	Document	228	5
BPMIE	Document	122	4
CFSIE	Document	43	98
UVIE	Document	105	–
Our-AdjuvantIE	Document	250	10

Table 7: **Comparison of scientific IE datasets in terms of task level, test set size, and number of fields.**

have a more noticeable effect on filling-time cost than on final extraction performance. Future work will include a full-scale ordering-stability study over the complete dataset to provide a more comprehensive assessment.

AdjuvantIE Dataset Sample


```
[
  {
    "Article_id": "Nature-18",
    "Adjuvant_id": "18-B",
    "Adjuvant_Name": "Alhydrogel",
    "Category": "Particle",
    "Sub_type": "aluminum salt",
    "Composition": "aluminum oxyhydroxide",
    "Morphology": "fibrous nanoparticle",
    "Particle_Size": "10 nm",
    "Particle_Structure": "semi-crystalline aluminum oxyhydroxide",
    "Target": "NLRP3 inflammasome",
    "Target_Cell": "macrophages",
    "Combination_mode": "adsorption"
  },
  {
    "Article_id": "Nature-37",
    "Adjuvant_id": "37-D",
    "Adjuvant_Name": "MF59",
    "Category": "Emulsion",
    "Sub_type": "oil-in-water",
    "Composition": "squalene, polysorbate 80 and sorbitan trioleate",
    "Morphology": "spherical droplet",
    "Particle_Size": "Not Mentioned",
    "Particle_Structure": "oil-in-water emulsion droplets",
    "Target": "Not Mentioned",
    "Target_Cell": "monocytes and dendritic cells",
    "Combination_mode": "Not-Mentioned"
  },
  {
    "Article_id": "Cell-101",
    "Adjuvant_id": "101-A",
    "Adjuvant_Name": "AS04",
    "Category": "Composite",
    "Sub_type": "aluminum salt",
    "Composition": "aluminum salt and monophosphoryl lipid A",
    "Morphology": "particle",
    "Particle_Size": "Not Mentioned",
    "Particle_Structure": "MPL adsorbed on aluminum salt",
    "Target": "TLR4",
    "Target_Cell": "antigen-presenting cells",
    "Combination_mode": "adsorption"
  },
  {
    "Article_id": "Cell-124",
    "Adjuvant_id": "124-A",
    "Adjuvant_Name": "CpG-1018",
    "Category": "Molecule",
    "Sub_type": "oligonucleotide",
    "Composition": "synthetic unmethylated DNA",
    "Morphology": "Not Mentioned",
    "Particle_Size": "Not Mentioned",
    "Particle_Structure": "Not Mentioned",
    "Target": "TLR9",
    "Target_Cell": "plasmacytoid dendritic cells",
    "Combination_mode": "mixing"
  }
]
```


Table 8: **Sample records from the AdjuvantIE benchmark.**

Model	Name	Formula	Acronym	Applications	Structure/Phase	Description	Overall macro-F1
SciIE SOTA Models							
LLM-NERRE (GPT-3)	0.668	0.664	0.613	0.715	0.608	0.494	0.627
Eunomia-Qwen2.5VL 7B	0.644	0.648	0.573	0.722	0.634	0.490	0.619
Mainstream LLMs							
GPT-5 Nano	0.701	0.651	0.622	0.749	0.641	0.477	0.640
Gemini2.5-flash	0.741	0.723	0.617	0.803	0.729	0.535	0.691
SudokuFill (Ours)							
SudokuFill-Qwen2.5VL 7B	**0.766**	**0.738**	**0.641**	**0.767**	**0.701**	**0.573**	**0.698**

Table 9: **Performance on the MaterialIE2 dataset.**

Model	Params	Entity-level F1	Row-level F1	Overall F1
Vanilla Backbones				
LLaVA-v1.5	7B	54.28	18.84	36.56
Phi3-Vision	7B	48.79	15.72	32.26
Intern-VL2.5	8B	60.11	21.91	41.62
SudokuFill Enhanced				
SudokuFill (LLaVA-v1.5)	7B	**60.27**	**20.32**	**40.30**
SudokuFill (Phi3-Vision)	7B	**53.88**	**16.11**	**35.00**
SudokuFill (Intern-VL2.5)	8B	**65.29**	**22.06**	**43.66**

Table 10: **Evaluation on Weak Baselines (AdjuvantIE benchmark)**

Metric	Value
Number of papers	50
Runs per paper	3
Papers with identical ordering across runs	39 (78%)
Papers with at least one ordering change	11 (22%)
Entity-F1 std across runs (11 papers)	67.09 ± 0.61
Row-F1 std across runs (11 papers)	25.47 ± 0.32

Table 11: **Ordering stability and its effect on extraction performance.**

Stage I: Page Agent Probing Prompt

```
You are a Page Agent for document-level scientific information extraction.

## Your role
- You are given:
  1. A single page `p` from a scientific PDF (text, tables, figures),
  2. A target field `f` with its definition
- Probe only this page to decide if it contains evidence for the target field.
- Be faithful to page evidence; do NOT guess.

## Target query
- Field: {field_name}
- Field definition: {field_definition}

## Evidence scope
- Use only evidence on this page: text spans, table cells, or figure regions.
- If not present, output `V = N/A` and `C = 0.0`.

## Confidence rubric (discrete levels)
- Use `C` in {0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0}.
- 0.0: No response / not applicable. The query is irrelevant to this page or no
candidate can be proposed;
output `N/A` and note "no mention on this page" in `E`.
- 0.1: Weak cue only. Some related terms appear but the target value is not stated;
evidence cannot support a concrete candidate (avoid proposing a value).
- 0.2: Slightly stronger cue but still cannot propose a concrete candidate;
evidence remains limited.
- 0.3: Ambiguous candidate. A citable location suggests a plausible candidate, but
the mention is implicit or non-unique and requires substantial inference; keep
below participation threshold.
- 0.4: Some evidence exists but still ambiguous; may require inference or
additional context.
- 0.5: Minimally grounded (participation boundary). A plausible candidate is
supported by verifiable localization, but it may be incomplete, non-unique, or
require light normalization; evidence exists but is not decisive.
- 0.6: Plausible candidate, partially verified; minor ambiguity remains.
- 0.7: Explicit but needs disambiguation. The value is explicitly stated with
verifiable evidence, yet multiple candidates are present or selection depends on
additional context.
- 0.8: Mostly unique and well-supported; minor ambiguity remains.
- 0.9: Explicit, unique, and well-supported. The value is clearly and uniquely
supported on this page, typically corroborated by multiple cues, with little room
for alternatives.
- 1.0: Decisive evidence. The value is unambiguous, uniquely specified in a
definitive form and fully consistent with all available cues.

## Output requirements
- Output a single JSON object with exactly three keys: `V`, `C`, `E`.
  * `V`: the proposed value (or `N/A`).
  * `C`: confidence score in the allowed set.
  * `E`: verifiable localization. Must include `page_id = {page_id}`. Also include
  one of:
    - `text_span`: an exact quote (short) OR start/end character offsets if
    available
    - `table_ref`: table_id + row header + column header + cell content
    - `figure_ref`: figure_id + caption snippet
- Keep `E` concise but checkable.

## Page content
{page_content}
```

Table 12: **Stage I Page Agent probing prompt.**

Stage I: Rank Agent Scheduling Prompt

```
You are a Rank Agent that schedules field-grounded queries for a document-level
extraction system.

## Your input
- A page-query matrix M that stores outputs (V_{p,q}, C_{p,q}, E_{p,q}) for each
page p and query q.
- Each query q corresponds to a schema field f with its field definition.
- You may be given lightweight field dependency hints.

## Your goal
- Output a prioritized query sequence pi for Stage II.
- The goal is to provide a stable starting point and reduce cascading errors by
handling easier, better-grounded queries earlier and leaving ambiguous queries
later.
- You must NOT modify any (V, C, E). You only schedule.

## How to reason (adaptive, not hard-coded)
- You may compute any summary statistics from M that help scheduling, for example:
  * MaxConf(q): the maximum confidence across pages for query q.
  * Part(q): the number of pages with C_{p,q} > 0.5 for query q.
- Interpret these signals cautiously: higher MaxConf often indicates stronger
evidence; Part reflects
how concentrated or dispersed the signal is; dependencies indicate which
fields can constrain others.
- There is no fixed rule for how to combine these signals. Instead, adaptively
balance them based on the observed evidence patterns in M.
- If dependencies conflict with raw evidence strength, you may reorder, but you
must explain why.

## Output format
- Output a JSON object with two keys:
  * pi: an ordered list of query ids (or field names) representing the execution
  order.
  * rationale: a short explanation (4-8 sentences) that justifies the ordering
  strategy, referencing evidence patterns in M (and dependency hints if used).

## Provided data
- Queries: {query_list}
- Dependency hints: {field_dependency}
- Page-query matrix summary: {matrix_summary}
```

Table 13: **Stage I Rank Agent scheduling prompt.**

Stage II: Page Agent Debate Prompt

```
You are a Page Agent in a multi-agent deliberation system for document-level
scientific information extraction.
You are assigned a specific page p (rendered as multimodal content: text, tables,
and figures).
Your task is to help resolve the current query q by proposing or revising a
candidate value using ONLY verifiable evidence on this page.

Inputs you will receive:
1) Query q = <field f, specification (f)>.
2) Current candidate pool C_pool: a list of candidates proposed so far for this
query (each with short evidence pointers).
3) History snapshot H_prev: brief summary of the previous round (leading candidate
and open objections).
4) (Optional) Hypothesis value V_hyp (only in re-thinking mode): treat it as a
hypothesis and search for support/contradiction on your page.

Rules:
- You must ground your response on this page only. Do not use outside knowledge.
- If you propose a value, it should be consistent with the field specification (f)
(format, units, allowed labels).
- If you cannot find a reliable signal on this page, return V = NA with low
confidence and explain why.
- You may revise your previous stance if you find new evidence or realize a
mismatch (entity, condition, unit, table row/column).
- Evidence E must be verifiable and localized.

Output format (strict JSON):
{
  "agent_role": "page",
  "page_id": "<p>",
  "value": "<V or NA>",
  "confidence": <C in [0,1]>,
  "evidence": {
    "page_id": "<p>",
    "evidence_type": "text|table|figure",
    "location": "brief pointer (e.g., paragraph index / table id + row/col /
    figure id + caption)",
    "quote_or_summary": "short excerpt or faithful summary (no long copying)"
  },
  "stance": "support|oppose|abstain",
  "notes": "if you revise, state what changed and why (e.g., unit mismatch
  corrected)."
}

Now produce your output for the current query.
```

Table 14: **Stage II Page Agent Debate Prompt.**

Stage II: Constraint Row Agent Prompt

```
You are the Row-View Constraint Agent in a multi-agent deliberation system.
Your job is NOT to propose new values. You only evaluate and comment on candidates
already proposed by Page Agents.

Inputs you will receive:
1) Query q = <field f, specification (f)>.
2) Current candidate pool C_pool: candidates for this query with evidence pointers.
3) Global Filling State G: a searchable state containing previously converged
records and fields (within and across documents).
4) History snapshot H_prev: brief summary of the previous round.

Your objective:
- Retrieve row-level regularities and constraints from G that are relevant to the
current query.
- For each relevant candidate in C_pool, decide whether it is supported
or challenged by row-level consistency and typical co-occurrence patterns.
- Select ONE candidate value V_sup from C_pool that you most strongly support or
challenge, and justify with evidence.

Rules:
- Do NOT invent new candidate values. V_sup must be chosen from C_pool.
- Use G only as referential guidance (patterns, consistency checks), not as ground
truth.
- Provide verifiable evidence pointers: cite what you retrieved from G (record ids
/ field names / matched patterns).
- If G provides no useful signal, return V_sup = "no_preference" with low
confidence.

Output format (strict JSON):
{
  "agent_role": "constraint_row",
  "supported_value": "<V_sup from C_pool OR no_preference>",
  "confidence": <C in [0,1]>,
  "evidence": {
    "source": "global_state",
    "retrieval_pointer": "what you retrieved (record ids / fields / pattern
    summary)",
    "constraint_type": "row_consistency|cooccurrence|canonical_pattern|exception",
    "summary": "why this supports or challenges V_sup"
  },
  "stance": "support|oppose|abstain",
  "notes": "state the key row-level constraint you used, and any caveat."
}

Now produce your output.
```

Table 15: **Stage II Constraint Row Agent Prompt.**

Stage II: Constraint Column Agent Prompt

```
You are the Column-View Constraint Agent in a multi-agent deliberation system.
You do NOT propose new values. You only evaluate candidates proposed by Page Agents.

Inputs:
1) Query q = <field f, specification (f)>.
2) Current candidate pool C_pool for this query.
3) Global Filling State G: previously converged fields/records (within and across
documents).
4) History snapshot H_prev.

Objective:
- Retrieve column-level priors: canonical value forms, alias normalization hints,
typical units/ranges (if numeric), and schema-consistent label sets.
- Check candidates in C_pool for format validity, alias consistency, and
compatibility with already resolved fields in the current document context.
- Select ONE candidate value V_sup from C_pool that you most strongly support or
challenge, and justify with retrieved signals.

Rules:
- V_sup must be chosen from C_pool (no new values).
- Treat G as referential, not absolute truth.
- If no useful signal exists, return V_sup="no_preference" with low confidence.

Output (strict JSON):
{
  "agent_role": "constraint_col",
  "supported_value": "<V_sup from C_pool OR no_preference>",
  "confidence": <C in [0,1]>,
  "evidence": {
    "source": "global_state",
    "retrieval_pointer": "retrieved canonical forms / aliases / unit or label
    constraints",
    "constraint_type": "format|alias|unit_range|label_set|dependency_check",
    "summary": "why this supports or challenges V_sup"
  },
  "stance": "support|oppose|abstain",
  "notes": "state normalization/constraint check and any exception case."
}

Now produce your output.
```

Table 16: **Stage II Constraint Column Agent Prompt.**

Stage II: Coordinator Agent Prompt

```
You are the Coordinator Agent (A_C) for multi-agent deliberation.
You must NOT introduce new values by yourself. Your role is to orchestrate rounds,
consolidate evidence, and decide whether to continue or stop.

Inputs you will receive at round t for query q:
- Query q = <field f, specification (f)>.
- Candidate pool C_pool^(t): list of candidate values with aggregated evidence
pointers and current support/opposition notes.
- Agent messages from this round: Page Agents output (V,C,E); Constraint Agents
output (V_sup,C,E).
- History memory H_prev = H^(t-1)(q): leading candidate and unresolved objections.
- Global filling state G (read-only during this query; write only after
convergence).
- A mode hint may be present: regular / anti-bias / re-thinking.

Your responsibilities:
1) Update the candidate pool:
   - Merge equivalent values (string/alias-level merge).
   - Attach new evidence to the corresponding candidate entry.
   - Track which agents support/oppose each candidate.
2) Update the round history H^(t)(q):
   - Record the new leading candidate and why (evidence-based).
   - Record unresolved objections and what evidence is missing to resolve them.
   - Record within-agent confidence changes (only compare each agent to itself
   across rounds).
3) Choose the next deliberation mode:
   - Regular debate by default.
   - Anti-bias if you detect early dominance plus persistent evidence-backed
   minority objection that is not answered.
   - Re-thinking if the current leading candidate relies on a single page source
   or a single agent.
4) Decide stop/continue:
   - Do NOT use absolute confidence comparisons across different agent types.
   - Declare convergence when the leading candidate stays unchanged across
   multiple rounds AND most agents only make marginal within-agent confidence
   updates AND no unresolved strong objection remains.

Output (strict JSON):
{
  "query_id": "<q>",
  "mode_next": "regular|anti_bias|re_thinking|stop",
  "leading_value": "<V*>",
  "supporting_evidence": ["<evidence pointers aggregated>"],
  "open_objections": ["<objection summaries or empty>"],
  "candidate_pool_updated": [
    {"value":"...", "supporters":[...], "opposers":[...], "evidence":[...]}
  ],
  "history_update": "compact H^(t)(q) summary",
  "converged": true|false,
  "final_output_if_converged": {
    "value":"<V*>",
    "evidence":["<E*> aggregated pointers"]
  }
}

If converged=true, provide final_output_if_converged and set mode_next="stop".
Otherwise set mode_next to the chosen deliberation mode for the next round.
```

Table 17: **Stage II Coordinator Agent Prompt.**