SciEvent: Benchmarking Multi-domain Scientific Event Extraction

Bofu Dong[1], Pritesh Shah[1], Sumedh Sonawane[1], Tiyasha Banerjee[1], Erin Brady[1]
Xinya Du[2], Ming Jiang[1,3]
[1]Indiana University Indianapolis, [2]University of Texas at Dallas, [3]University of Wisconsin-Madison
bofudong@iu.edu ming.jiang@wisc.edu

Abstract

Scientific information extraction (SciIE) has primarily relied on entity-relation extraction in narrow domains, limiting its applicability to interdisciplinary research and struggling to capture the necessary context of scientific information, often resulting in fragmented or conflicting statements. In this paper, we introduce SciEvent[1], a novel multi-domain benchmark of scientific abstracts annotated via a unified event extraction (EE) schema designed to enable structured and context-aware understanding of scientific content. It includes 500 abstracts across five research domains, with manual annotations of event segments, triggers, and fine-grained arguments. We define SciIE as a multi-stage EE pipeline: (1) segmenting abstracts into core scientific activities—*Background*, *Method*, *Result*, and *Conclusion*; and (2) extracting the corresponding triggers and arguments. Experiments with fine-tuned EE models, large language models (LLMs), and human annotators reveal a performance gap, with current models struggling in domains such as sociology and humanities. SciEvent serves as a challenging benchmark and a step toward generalizable, multi-domain SciIE.

1 Introduction

Scientific information extraction (SciIE) distills structured knowledge from unstructured scientific articles and supports key scientific applications such as literature review (Hong et al., 2021), paper recommendation (Ikoma and Matsubara, 2023), and knowledge discovery (Stavropoulos et al., 2023), especially in recent years as many domains are facing a publication deluge.

Existing works on SciIE generally follow an entity-relation extraction (ERE) paradigm that aims to extract isolated scientific concepts and connect them by identifying semantic relations, either binary (Luan et al., 2018; Zhang et al., 2024) or N-ary (Jain et al., 2020; Zhuang et al., 2022). Despite remarkable contributions made by prior studies, one major concern is that representing scientific content as disconnected entity-relation tuples may fragment the underlying narrative and even introduce conflicting statements, especially when synthesizing information across multiple publications. As shown in Figure 1, one paper may generate the tuple ⟨ "GPT-3.5-Turbo", "better than", "GPT-4-Turbo"⟩, while another produces the opposite. Lacking contextual cues such as task setup or evaluation criteria, these tuples alone fail to convey meaningful or reliable scientific insights.

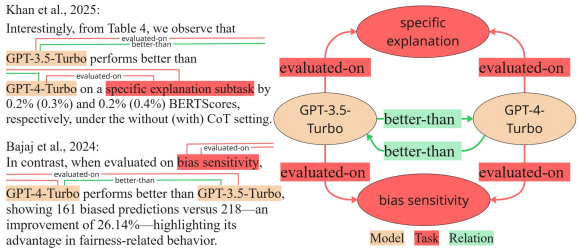

Figure 1: Conflicting statements in entity-relation extraction. ⟨GPT-3.5-Turbo, better than, GPT-4-Turbo⟩ vs. ⟨GPT-4-Turbo, better than, GPT-3.5-Turbo⟩

Inspired by the heavily context-dependent nature of scientific publications, we adopt an event extraction (EE) paradigm. This paradigm focuses on identifying triggers that best represent each event and extracting associated arguments, which are then assigned specific semantic roles. This enables a more structured and context-aware representation of important scientific information. Despite its potential for representing scientific information, a major limitation of existing EE efforts in the scientific domain is their narrow focus on specific fields, often resulting in the development of domain-specific EE schemas. For example, Zhang et al. (2024) and Jain et al. (2020) focus on machine learning, and Kim et al. (2011) focus on bio-molecule area. Given the rapid growth of interdisciplinary research in

[1]Our code and benchmark are released at https://github.com/desdai/SciEvent.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 17225–17255
November 4-9, 2025

recent years (Leto et al., 2024; Okamura, 2019), there is an increasing need for a unified scientific EE schema capable of generalizing across diverse scholarly domains.

To address this gap, we introduce SciEvent, a unified EE schema for scientific texts, along with a dataset featuring manually annotated events and fine-grained arguments drawn from diverse research abstracts. Building on this dataset, we define three SciIE tasks: (1) event segmentation, which involves dividing the text into spans that represent core scientific activities such as *Background*, *Method*, *Result*, and *Conclusion*; (2) trigger identification, which aims to detect the key anchor of each scientific event; and (3) argument extraction, which focuses on identifying the arguments involved in each scientific activity and assigning them roles such as context, method, or result. Differing from conventional EE pipelines, we introduce event segmentation as a preliminary task, recognizing that events in scientific texts often span multiple sentences and lack clear boundaries. Additionally, trigger words in scientific texts—such as "show", "demonstrate", or "present"—are frequently shared across different event types. Without first segmenting the text into discrete events, it becomes challenging to accurately delineate event boundaries, increasing the risk of misinterpreting or misclassifying both triggers and their associated arguments.

SciEvent contains 500 abstracts from five diverse scientific domains, each fully annotated using an EE paradigm. To evaluate the challenges posed by this dataset, we assess the performance of fine-tuned EE models, tuning-free large language models (LLMs), and human annotators. The results demonstrate SciEvent's broad domain coverage and reveal that existing models consistently lag behind human performance. This gap highlights the limitations of current approaches and the absence of EE models capable of generalizing across scientific domains.

2 Related Work

Event Extraction Existing work on event extraction (EE) typically frames the task via two paradigms. One is trigger-argument extraction (Walker et al., 2006; Hsu et al., 2022; Lin et al., 2020), where the trigger serves as the event anchor, most clearly signaling the occurrence of an event, while the arguments represent entity mentions that participate in the event, each fulfilling distinct roles. The other one treats EE as a trigger-free template-filling task (MUC, 1992; Du and Cardie, 2020a; Huang et al., 2021), aiming to extract event-relevant arguments and assigning them to specific roles within each event template. The latter mainly focuses on document-level EE (Du and Cardie, 2020a), while the former has been widely used in both sentence-level (Walker et al., 2006) and document-level EE (Li et al., 2021). Our benchmark follows the trigger-argument paradigm.

Regarding EE benchmarks, prior studies have largely focused on data in generic domains. Popular examples include newswire (Grishman and Sundheim, 1996; Nguyen et al., 2016; Doddington et al., 2004; Ebner et al., 2020; Song et al., 2015), Wikipedia (Li et al., 2021; Pouran Ben Veyseh et al., 2022), social media (Sharif et al., 2024; Wang and Zhang, 2017; Comito et al., 2019) and widely-used knowledgebases like FrameNet (Baker et al., 1998) and PropBank (Bonial et al., 2014). While some researchers have broadened the scope of EE to scientific literature, their efforts tend to center the biomedical domain, particularly emphasizing state changes and interactions between biomolecules such as genes and proteins (Kim et al., 2011; Pyysalo et al., 2012; Kim et al., 2013). Differing from prior work, we extend EE to encompass a broader range of scientific domains, creating a unified annotation schema designed to facilitate interdisciplinary information extraction.

Scientific Information Extraction Research on scientific information extraction (IE) primarily targets two main types of information: (1) citation-based analysis, which involves identifying either binary citation influence classification (Kunnath et al., 2020; N. Kunnath et al., 2021; Maheshwari et al., 2021) or multi-class citation intents (purpose) classification (Cohan et al., 2019; Jurgens et al., 2018), and (2) content-based analysis (Gupta and Manning, 2011; Tsai et al., 2013; Gábor et al., 2016; Pronesti et al., 2025), which primarily focuses on extracting scientific entities, supporting evidence, and semantic relationships among them, with the ultimate goal of building concept-centric knowledge graphs (Ma et al., 2022; Zhang et al., 2020; Sap et al., 2019). For example, SciERC (Luan et al., 2018), consists of 500 scientific abstracts annotated with scientific entities, their pairwise relations, and coreference clusters. SciREX (Jain et al., 2020) provides annotations across 438 full documents, covering four entity types: TASK,

DATASET, METHOD, and METRIC. Beyond general knowledge extraction, some studies further focus on specific research subjects. This line of work designs domain-specific event extraction tasks to capture fine-grained scientific activities (He et al., 2024,Kim et al., 2011, Huang et al., 2020, Björne et al., 2010). For example, various biomedical EE tasks have been proposed to investigate biological processes such as protein-protein and gene-disease interactions (Kim et al., 2013; Kim et al., 2011; Björne et al., 2010). Our work similarly focuses on scientific EE. However, differing from prior works targeting specific domain, we aim to design a unified schema for organizing general scientific activities across diverse scientific domains.

3 SciEvent Benchmark

Data Collection To support cross-domain evaluation and capture diverse writing conventions, we select publicly available, peer-reviewed scientific abstracts published in 2023 to reflect contemporary language use. We select five domains: natural language processing (NLP) from the Annual Meeting of the Association for Computational Linguistics (ACL) (Association for Computational Linguistics, 2023), social computing (SC) from the Proceedings of the ACM on Human-Computer Interaction (CSCW) (Association for Computing Machinery, 2023), medical informatics (MI) from the Journal of Medical Internet Research (JMIR) (JMIR Publications, 2023), computational biology (CB) from the Bioinformatics (Oxford University Press, 2023), and digital humanities (DH) from the Digital Humanities Quarterly (Alliance of Digital Humanities Organizations, 2021–2023)[2].

These domains are selected for their methodological diversity, resource availability, relevance to interdisciplinary research, and representativeness of their respective fields. NLP and CB domains are well-studied and offer structured, technical abstracts, while SC and DH are underrepresented and characterized by more narrative, context-rich writing. To support document-level modeling, we retain abstracts with at least three sentences and two identifiable events, filtering out those that are too short to provide meaningful structure. In total, we collect 500 scientific abstracts—100 each in NLP, SC, and CB, 120 in DH, and 80 in MI. DH has fewer publications and shorter abstracts, so we extend the sampling range to 2021—2023 and include 120 abstracts to ensure sufficient coverage of domain variation. MI abstracts are longer and denser, so we select 80 abstracts to balance event content comparability across all five domains. We conduct a detailed keyword analysis based on each domain's call for papers to ensure comprehensive coverage and minimize bias in our dataset. Additional details are provided in Appendix G

Annotation Pipeline Overall, our annotation pipeline consists of two stages: (1) event segmentation, and (2) trigger-argument extraction. In the first stage, we segment an abstract into four event types: *Background*, *Method*, *Result*, and *Conclusion*, which are adopted from the most common aspects of scientific publications (U.S. National Library of Medicine, 2023). In the second stage, we further annotate each segment at a fine-grained level, focusing on identifying the event trigger and role-specific arguments.

In prior event extraction works, particularly in newswire and broadcast domains, triggers like "attack" define clear and stable event frames, with roles such as "attacker" and "target" naturally grounded in the trigger's semantics. In scientific texts, however, single-word triggers like "show" lack this clarity. Even after event segmentation and the event type (e.g., "Result") is known, the trigger alone does not specify what the event is about. Roles like "people who show" or "shown item" are not meaningful on their own, as the event's meaning depends on the full proposition. For example, "showing a promising result" differs from "showing a methodological limitation". With this consideration, we represent the trigger as a tuple of ⟨Agent, Action, Object⟩, anchoring the event in its core semantics. Notably, our empirical investigation on raw data shows that in some cases, the object in an event trigger may consist of two non-contiguous text spans. For example, "protein sequences" and "gene expression profiles" in a "Method" event: "We analyze protein sequences, which exhibit structural variation, and gene expression profiles . . . " are the objects of the action "analyze". Accordingly, we specify the labels *Primary Object* and *Secondary Object* for annotation. When we have two annotated object spans in an event, we concatenated them for further analysis.

Given the trigger identified per event, we then annotate its relevant arguments. We define nine argument roles: *Context*, *Purpose*, *Method*, *Result*, *Analysis*, *Challenge*, *Ethical*, *Implication*, and *Con-*

[2]Full source attributions are included in the benchmark metadata available in our public GitHub repository.

tradiction. Each role targets a specific dimension of scientific abstract, adapted from Core Scientific Concepts (Liakata et al., 2012) and inspired by scientific writing guides (Paltridge, 2002; Alley, 1996). While some argument roles share names with event types (e.g., Method, Result), they are not restricted to those events. For example, evaluation method often appear within the Result event. We attach the detailed codebook in Appendix B.

Annotation Quality We employ five graduate students as annotators, all specializing in NLP and are either native English speakers or PhD students. Each annotator has domain expertise in at least one of the five selected fields, ensuring comprehensive coverage across all diverse scientific areas. To evaluate the quality of event segment annotations, we randomly sample 10 abstracts per domain and had two annotators independently annotate each. Intercoder reliability, measured by Cohen's Kappa (Cohen, 1960), is 0.83, showing strong agreement.

Considering that trigger and argument extraction involve more fine-grained and complex annotations than event segmentation, it increases the likelihood of annotator disagreement. To ensure consistency, we employ a collaborative, multi-round, discussion-based annotation process (Oortwijn et al., 2021) rather than a single-pass approach. Annotators first label the data independently, followed by review sessions with a meta-annotator to enforce codebook alignment. This cycle is repeated over six rounds, yielding 100% agreement on triggers and 95.41% agreement (4703/4929) on arguments. The remaining 4.59% are resolved through majority voting among all five annotators, resulting in full team consensus on the final annotations. Notably, all disagreements in trigger and argument annotations, such as span variations and ambiguous argument roles, are resolved through specific rules outlined in the Codebook (Appendix B.4). To assess human performance for comparison with models, we additionally recruit six untrained annotators to independently annotate a randomly selected subset of our benchmark (consisting of 75 abstracts). For these annotators, we only provide them with a brief task description and basic instructions for using the annotation interface (see appendix B.1).

Data Analysis Using the above annotation pipeline, we construct a dataset of 500 annotated scientific abstracts containing 8,911 structured mentions, as shown in Table 1. Its broad domain coverage supports robust cross-domain analysis.

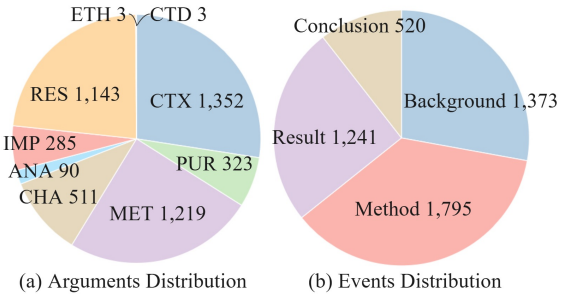

Figure 2: Distribution of (a) argument roles and (b) event types across the dataset.

As shown in Figure 2, the most frequently annotated arguments are Context (CTX), Method (MET), and Result (RES), highlighting the dataset's emphasis on core components of scientific reporting. Rare arguments such as Contradictions (CTD) and Ethical (ETH) suggest that such aspects are rarely discussed in the abstracts. The most common event type is the Method, consistent with typical abstracts structures. Moreover, Figure 3 shows that argument types align well with event types. For example, Context appears predominantly in Background events, supporting the reliability and internal consistency of our annotations.

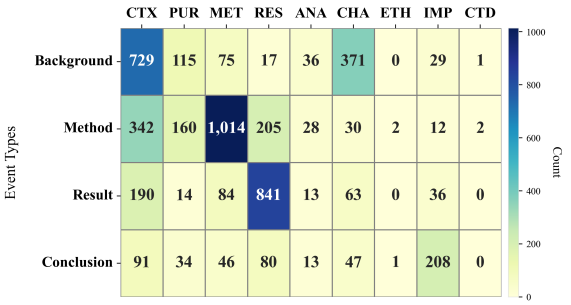

Figure 3: Distribution of argument across event types

4 Task Definition

Given a document represented as a sequence of sentences $D = \{s_1, \ldots, s_N\}$, our goal is to extract a set of scientific events $E = \{e_1, \ldots, e_M\}$, where each event e_i is a tuple defined as: $e_i = \langle s_{ij}, type_i, \text{Trigger}_i, \text{Arg}_i \rangle$, where s_{ij} denotes a contiguous sentence span in D, $type_i$ is the event type, Trigger_i is the event trigger and Arg_i consists of event arguments involved in e_i. Specifically, we define Trigger_i as an agent-action-object tuple:

$$\text{Trigger}_i = \langle \sigma_{\text{agent}},\ \sigma_{\text{action}},\ \sigma_{\text{object}} \rangle,$$

where $\sigma \in D$ is a token span that specifies who-does-what in e_i, respectively. We further define Arg_i as a list of argument-role pairs:

Dataset	#Doc	#Mentions	Arg./Ent. Types	Avg Sent./Evt	Paradigm	Source	Domains
SciREX	438	8,592	4	-	ERE	Full paper	ML
SciERC	500	8,089	6	-	ERE	Abstract	Speech, ML, CV, AI
SemEval17	493	8,529	3	-	ERE	Paragraph	CS, MS, Physics
SemEval18	500	7,505	1	-	ERE	Abstract	CL
SciER	106	24,518	3	-	ERE	Full paper	ML
Genia2011	1,224	21,549	10	1	EE	Abstract/Full	BioMol
SciEvent (ours)	500	8,911	9	2.95	EE	Abstract	NLP, SC, CB, MI, DH

Table 1: Comparison of scientific IE datasets. Abbreviations: **Arg./Ent. Types** = Argument/Entity Types, **Avg Sent./Evt** = Average Sentence Per Event, **NLP** = Natural Language Processing, **SC** = Social Computing, **CB** = Computational Biology, **MI** = Medical Informatics, **DH** = Digital Humanities, **ML** = Machine Learning, **AI** = Artificial Intelligence, **CV** = Computer Vision, **CS** = Computer Science, **MS** = Material Science, **CL** = Computational Linguistics, **BioMol** = Biomolecular.

$$\text{Arg}_i = \{a_{ij}, r\},$$

where $a_{ij} \in D$ denotes the token span that refers to a participating argument entity, and r is the specific role that the argument a_{ij} plays in e_i.

To achieve our goal on SciEvent, we define three tasks: (1) event segmentation, (2) trigger identification, and (3) argument extraction. The details of each task are described below.

Task 1: Event Segmentation This task aims to segment any given document D into contiguous sentence spans $\{s_{ij}\}$, with each span s_{ij} corresponding to an event classified under one of four scientific event types $type_i$.

We evaluate model predictions using *Exact Match (EM)* and *Intersection over Union (IoU)* metrics, adapted from span-based evaluation metrics in SemEval (Segura-Bedmar et al., 2013) and MUC-5 (Chinchor and Sundheim, 1993). For each predicted event segment $(\hat{s}_{ij}, \hat{type}_i)$, the above metrics are defined as follow:

- **Exact Matching (EM):** $\hat{s}_{ij} = s_{ij}$ and $\hat{type}_i = type_i$
- **Intersection over Union (IoU):** $\frac{|\hat{s_{ij}} \cap s_{ij}|}{|\hat{s_{ij}} \cup s_{ij}|} > 0.5$ and $\hat{type}_i = type_i$.

For both strategies, we report Precision (P), Recall (R), and F1-score (F1) over the set of predicted and gold event segments.

Task 2: Trigger Identification This task focuses on extracting the trigger for each detected event. As this is a document-level task and scientific events often include multiple candidate triggers, we handle this step separately to explicitly evaluate the model's ability to correctly identify the core semantic components of an event once its span and type have been identified.

For evaluation, we concatenate each trigger tuple's three components and compute macro ROUGE-L (Lin, 2004) between predicted and annotated triggers. Given that ROUGE-L measures the longest common subsequence overlap, we believe that this metric can capture both lexical similarity and structural alignment

Task 3: Argument Extraction We decompose this task into two sub-tasks:

- **Argument Identification (Arg-I):** Predict the set of argument entity spans $\{a_{ij}\}$ per event.
- **Argument Classification (Arg-C):** Predict the semantic role r associated with each identified argument span a_{ij}.

We evaluate Arg-I using the F1 score based on the span-matching strategies described in Task 1: *EM* and *IoU* (with a threshold of 0.5). For Arg-C, a prediction is considered correct only if it both matches the gold argument span and also assigns the correct argument role.

5 Experiment Settings

Prompting-based LLM baselines We consider four state-of-the-art LLMs as baseline models, including: (1) meta-Llama-3.1-8B-Instruct (Llama) (Meta AI, 2024), (2) Qwen2.5-7B-Instruct (Qwen) (Qwen Team, 2024), (3) DeepSeek-R1-Distill-Llama-8B (DS-R1-Llama) (DeepSeek-AI, 2025), and (4) GPT-4.1 (GPT) (OpenAI, 2025). For Task 1, we conduct a preliminary study on prompt design under the zero-shot manner and used the best prompt adapted from Sharif et al., 2024. For Task 2 and 3, we design the prompt template based on a preliminary analysis of existing prompting strategies, including metacognitive prompting (Wang and Zhao, 2024), instruction-based prompting (Sharif et al., 2024) and paraphrasing these

prompts. Considering the risk of model performance being sensitive to the number of examples in the prompt, we test prompts with 0 to 5 examples. We finally employ the prompt adapted from Sharif et al., 2024, which shows consistently better performance across multiple trials. Notably, the prompt content (see details in Appendix C) is derived from our annotation codebook, to ensure a fair comparison between LLMs and humans.

Tuning-based baselines In addition to prompting-based baselines, we follow prior studies (Huang et al., 2024; Tong et al., 2022) and also employ turning-based models on our event extraction tasks, adopting three state-of-the-art approaches: (1) DEGREE (Hsu et al., 2022), a data-efficient generative approach to event argument extraction that leverages prompt-based learning for better generalization. (2) OneIE (Lin et al., 2020), a joint information extraction framework that simultaneously performs entity, relation, and event extraction using a unified representation. (3) EE_QA (Du and Cardie, 2020b), a transformer-based model that frames information extraction as a question-answering task, enabling contextualized argument extraction. For all three models, we follow the splitting practice used in prior work (Huang et al., 2024) and adopt the same approach and split the training, development, and test sets by document with a ratio of 80%, 10%, and 10%.

6 Experiment Results

Model	EM			IoU		
	P	R	F1	P	R	F1
DS-R1-Llama	31.26	34.13	32.63	58.97	64.38	61.56
Qwen	43.51	36.30	39.58	70.30	58.65	63.95
Llama	38.67	31.70	34.84	62.04	50.85	55.89
GPT	**59.07**	**62.96**	**60.95**	**82.98**	**88.45**	**85.63**

Table 2: Scientific event segmentation performance (%) on zero-shot LLMs using Exact Match (EM) and Intersection over Union (IoU) metrics, showing Precision (P), Recall (R), and F1-score

Scientific event segmentation Table 2 shows the results of LLM performance under zero-shot manner. We observe that GPT clearly outperforms all others by a wide margin, achieving 60.95% F1 under EM and 85.63% under IoU, indicating its strong ability to identify and segment coherent scientific spans. Qwen ranks second, while Llama and DS-R1-Llama trail closely with modest differences. These results suggest that segmentation is best handled by higher-capacity models like GPT.

Methods	P	R	F1
Tuning-based models			
EEQA	**81.93**	34.57	45.05
DEGREE	64.56	63.49	56.85
OneIE	73.73	**79.40**	72.40
Zero-shot LLMs			
DS-R1-Llama	29.12	27.10	26.74
Qwen	43.84	55.25	47.57
Llama	54.88	61.07	55.83
GPT	65.38	72.73	67.57
One-shot LLMs			
DS-R1-Llama	41.81	41.94	40.72
Qwen	56.17	68.48	59.98
Llama	53.08	63.83	56.45
GPT	72.67	77.77	74.05
Two-shot LLMs			
DS-R1-Llama	34.59	36.21	34.29
Qwen	57.27	69.71	61.18
Llama	58.94	61.18	58.34
GPT	73.38	78.45	74.76
Five-shot LLMs			
DS-R1-Llama	38.63	35.63	35.18
Qwen	57.43	66.88	60.18
Llama	32.05	32.83	31.37
GPT	73.70	78.82	**75.08**

Table 3: ROUGE-L scores (%) for baseline models on the SciEvent trigger identification task, showing Precision (P), Recall (R), and F1.

Trigger Identification Table 3 displays the models' performance on trigger identification. GPT (five-shot) achieves the best result (F1: 75.08%), while OneIE also performs competitively (F1: 72.40%). EEQA exhibits extremely high precision (P: 81.93%) but poor recall (R: 34.57%), suggesting over-conservative predictions. Across all LLMs, one-shot prompting consistently improves performance, with DS-R1-Llama showing the largest gain (F1: +13.98%). While the improvement from zero-shot to one-shot is substantial, surprisingly, further adding more examples yields at most a 1% gain and can sometimes even reduce performance, particularly for Llama and DS-R1-Llama. Our observation aligns with prior findings that in-context learning may amplify reliance on superficial patterns in demonstrations (Min et al., 2022), which is insufficient for the fine-grained comprehension required by event extraction. Accordingly, our subsequent analysis focuses on zero-shot and the overall best-performing one-shot prompts.

Methods	Arg-I (IoU)			Arg-C (IoU)		
	P	R	F1	P	R	F1
Tuning-based models						
EEQA	32.09	33.77	32.91	25.85	27.20	26.51
DEGREE	**67.79**	19.13	29.84	**48.99**	13.83	21.57
OneIE	51.11	**56.29**	**53.57**	39.69	**43.71**	**41.61**
Zero-shot LLMs						
DS-R1-Llama	31.11	16.46	21.53	16.32	8.63	11.29
Qwen	35.68	26.41	30.35	17.58	13.01	14.96
Llama	24.37	24.90	24.63	11.68	11.93	11.80
GPT	43.03	55.56	48.50	30.40	39.25	34.26
One-shot LLMs						
DS-R1-Llama	42.62	17.67	24.98	19.59	8.12	11.48
Qwen	46.33	30.36	36.69	20.96	13.74	16.60
Llama	44.70	34.08	38.68	18.93	14.44	16.38
GPT	50.14	50.22	50.18	34.60	34.66	34.63
Two-shot LLMs						
DS-R1-Llama	42.66	20.01	27.24	14.37	6.74	9.18
Qwen	46.16	31.43	37.39	21.08	14.35	17.08
Llama	40.87	25.11	31.11	18.10	11.12	13.78
GPT	49.12	51.29	50.18	33.99	35.49	34.72
Five-shot LLMs						
DS-R1-Llama	36.31	20.92	26.55	13.62	7.85	9.96
Qwen	46.94	31.36	37.60	21.67	14.48	17.36
Llama	38.36	8.93	14.49	14.98	3.49	5.66
GPT	50.04	49.93	49.98	34.51	34.42	34.47

Table 4: IoU-based Precision (P), Recall (R), and F1-score (%) on baseline models for argument identification (Arg-I) and classification (Arg-C) tasks.

Argument Extraction Table 4 reports the performance of all baselines on argument extraction in SciEvent. OneIE achieves the highest scores (Arg-I: 53.57%, Arg-C: 41.61%), benefiting from its global features and constraints. DEGREE shows high precision but low recall, indicating that it often misses relevant arguments in scientific abstracts. Among LLMs, GPT (two-shot) performs best (Arg-I: 50.18%, Arg-C: 34.72%), while other models perform notably worse, especially on argument classification (Arg-C around 15%). One-shot prompting provides a modest gain over zero-shot settings, whereas adding more in-context examples shows similar diminishing returns observed in trigger identification. This indicates that merely increasing the number of few-shot examples is insufficient to overcome the fine-grained challenges of scientific argument extraction.

Human performance We compare model and human performance on argument classification. We do not report results for event segmentation, as the Cohen's kappa score of 0.83 (exact match) on a subset indicates consistently high agreement among annotators, suggesting that event segmentation is relatively unambiguous for humans. As shown in Figure 4, there is a substantial gap between human performance and the best model (20%). This highlights the challenge of multi-domain scientific event extraction and the value of SciEvent for advancing argument level scientific event extraction.

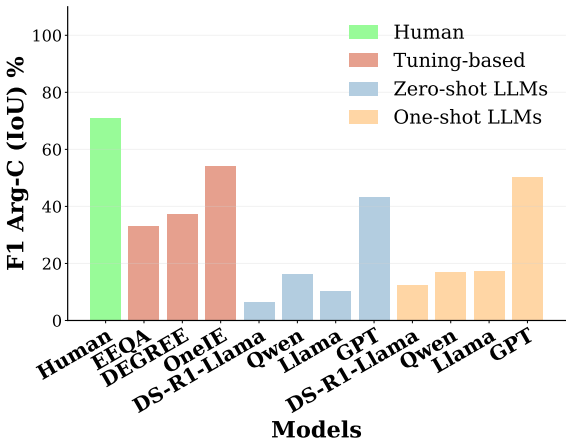

Figure 4: Human performance compared to all baselines on argument classification (Arg-C) using IoU F1 scores.

What is the impact of argument type on argument classification? Figure 5 displays IoU-based F1 scores for argument classification across argument roles. Among tuning-based and LLM-based models, OneIE and GPT achieve the strongest performance across nearly all argument roles. Qwen achieves a spike on Contradiction, due to a few correct extractions, but shows worse performance overall. Across all models, Challenge, Result, and Method yield the highest F1 scores, due to their clearer lexical cues and more regular positioning in scientific abstracts. In contrast, arguments like Ethical, Contradiction, and Analysis remain challenging due to data sparsity and a lack of consistent lexical patterns.

What is the impact of event type on argument classification? The arguments in Method exhibit a notable gap: strong performance with supervision (OneIE, EEQA) but poor with zero-/one-shot LLMs on argument classification task (Figure 6). This finding suggests that arguments in the Method events are most demanding, due to event's complex structure, arguments' varied phrasing, and dependence on technical details, making performance poorer without supervision. Furthermore, Conclusion shows the lowest Arg-C performance for most models. EEQA performs better because its QA-based templates help extract the implicit and interpretive content typical of Conclusion events.

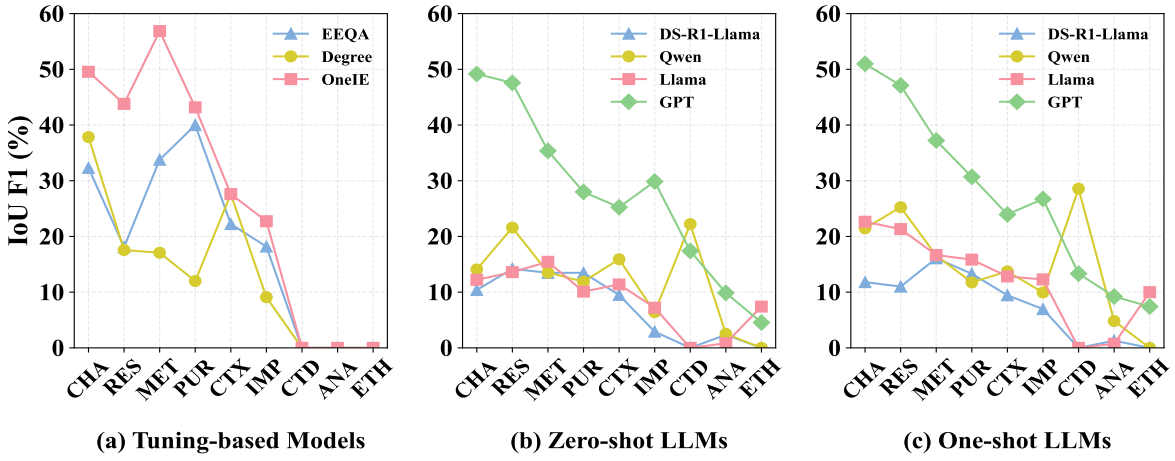

Figure 5: Intersection-over-Union (IoU) on Arg-C F1-scores (%) across different argument roles for various models on Analysis (ANA), Challenge (CHA), Context (CTX), Method (MET), Purpose (PUR), Result (RES), Ethical (ETH), Implication (IMP), Contradictions (CTD).

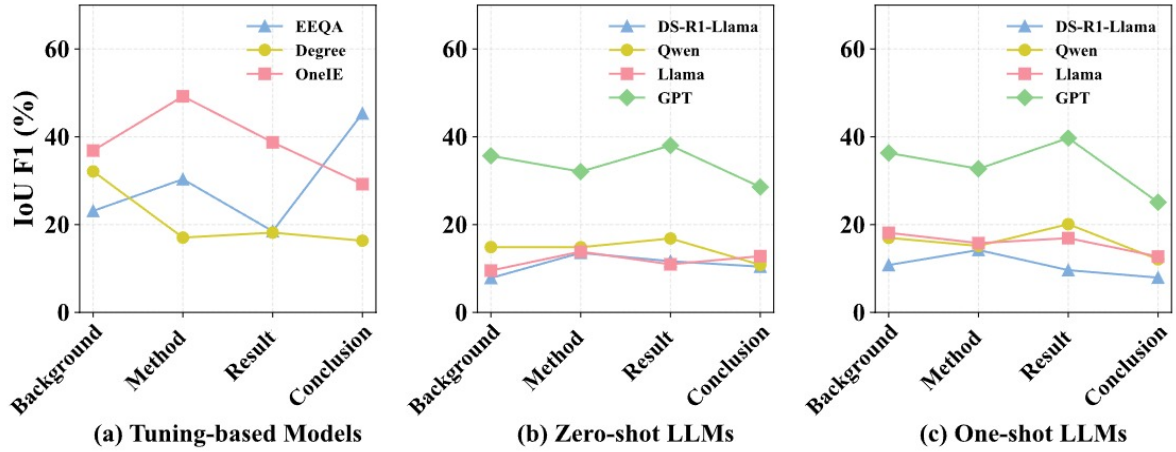

Figure 6: Comparison of Intersection-over-Union (IoU) on Arg-C F1-scores (%) across different event types for various models on *Background*, *Method*, *Result*, and *Conclusion* events.

To examine whether event type awareness can improve argument extraction for LLMs, we experiment with incorporating event type information into the prompts. Specifically, we explore two strategies: (1) providing the true event type directly in the prompt, and (2) asking the LLM to first predict the event type and then proceed with argument extraction. Both strategies outperform the original prompt, which lacks event type information, by approximately 2 to 4% (Table 5), suggesting that event type awareness enhances LLMs' performance on our benchmark.

What is the impact of scientific domains on argument classification? In the argument classification task (Figure 7), Natural Language Processing and Computational Biology domains yield the highest F1 scores, benefiting from consistent linguistic patterns and clearer argument structures. In contrast, Digital Humanities and Medical Informatics present greater challenges, due to varied rhetorical styles and longer, denser abstracts, respectively.

How does removal of domain affect performance? We compare the argument classification performance of the OneIE model under the Exact Match (EM) setting using the full training set versus ablated training sets (Figure 8). Removing a domain from training data leads to a noticeable drop in its corresponding performance, confirming that domain-specific knowledge contributes directly to accurate argument classification. The largest declines are observed in Digital Humanities and Computational Biology, indicating that these domains contain more unique or specialized linguistic patterns that are not easily generalized from other domains. In contrast, Medical Informatics shows relatively smaller drop, suggesting better generalizability or partial overlap with language patterns present in the other domains.

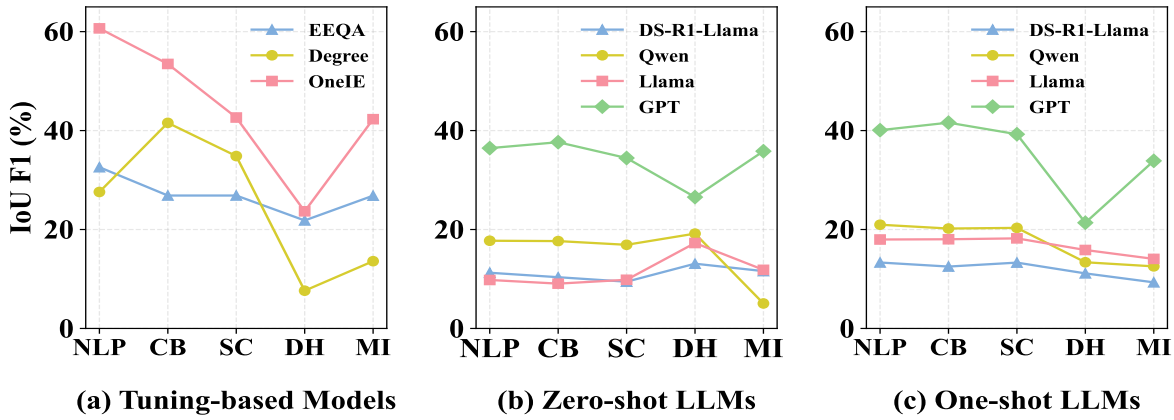

Figure 7: Comparison of Intersection-over-Union (IoU) on Arg-C F1-scores (%) across different academic domains for various models on Natural Language Processing (NLP), Computational Biology (CB), Social Computing (SC), Digital Humanities (DH), and Medical Informatics (MI).

Methods	Arg-I (IoU)			Arg-C (IoU)		
	P	R	F1	P	R	F1
True Event-Type LLMs						
DS-R1-Llama	27.25	17.95	21.64	16.50	10.87	13.10
Qwen	31.92	35.81	33.75	16.74	18.78	17.70
Llama	17.51	27.90	21.51	9.41	14.99	11.56
GPT	42.12	**57.13**	48.49	31.29	**42.44**	36.02
Pred Event-Type LLMs						
DS-R1-Llama	28.29	17.61	21.70	17.12	10.65	13.13
Qwen	31.93	35.28	33.52	17.17	18.97	18.02
Llama	20.53	28.71	23.94	10.36	14.48	12.08
GPT	**44.42**	55.22	**49.23**	**32.51**	40.42	**36.04**

Table 5: IoU-based Precision (P), Recall (R), and F1-score (%) comparing argument identification (Arg-I) and classification (Arg-C) performance given true event type or predicted event type.

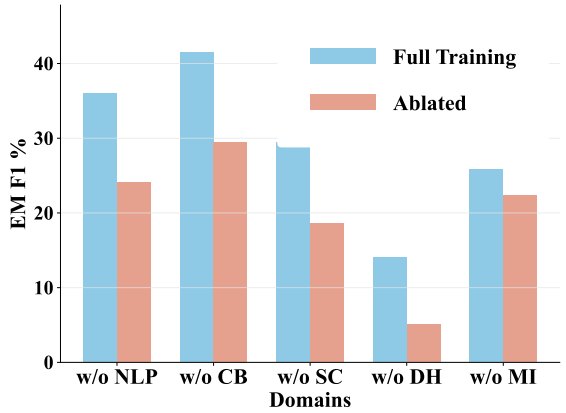

Figure 8: Arg-C F1-scores reported for full training versus training with one domain removed for OneIE under Exact Match (EM).

7 Conclusion

In this paper, we introduce SciEvent, a novel benchmark for SciIE across multiple domains. By framing scientific texts as a sequence of universal events and corresponding fine-grained arguments, SciEvent provides a unified and domain-independent structure for representing scientific information. Specifically, We develop an annotation pipeline comprising event segmentation and trigger-argument extraction, and defined three corresponding tasks: (1) event segmentation, (2) trigger identification, and (3) argument extraction. Our benchmark covers five diverse domains with manual annotations, enabling robust evaluation of Event Extraction. Experiments on diverse state-of-the-art tuning-based Event Extraction systems and tuning-free LLMs show clear performance gaps (∼20%) between model predictions and human annotations, especially on argument classification task. SciEvent supports applications such as knowledge graph construction, cross-domain literature review, and scientific summarization. It provides a challenging testbed for extracting nuanced scientific information, benefiting both NLP researchers for advancing event extraction methodology and evaluating cross-domain generalization, and interdisciplinary scholars for accelerating literature review, synthesizing findings, and generating domain knowledge resources.

Limitations

One limitation of our work is the potential for data contamination in large language models, as our dataset is constructed from recent publications (mostly from 2023, and 2021 to 2023 for Digital Humanities), which may overlap with LLM pretraining corpora. Nevertheless, we emphasize that our benchmark offers a novel formulation by representing scientific abstracts as structured sequences of events, enabling a context-aware capture of key scientific information. This event-centric SciIE schema is novel, and current LLMs lack training to extract scientific content in this structured manner. Additionally, SciEvent is built on abstracts only, which, while concise and widely available, may omit key discourse elements found in full papers limiting applicability to document-level information extraction. In future work, we plan to extend SciEvent to include full papers to better support comprehensive scientific IE, and also consider more event types and arguments roles since the full paper can contain more information such as Assumptions.

Ethical Considerations

We provide details about compensation rate for annotators. We recruited eleven graduate students in total and provided a compensation rate of $12.80 per hour. This rate applied to both gold-standard annotation and human performance baseline annotations.

Acknowledgement

We thank the anonymous reviewers for their insightful comments and helpful suggestions. We are also grateful to all annotators for their contributions to this work. Finally, we acknowledge the support of the Institute of Integrative Artificial Intelligence at Indiana University.

References

Michael Alley. 1996. *The Craft of Scientific Writing*, 3rd edition. Springer, New York.

Alliance of Digital Humanities Organizations. 2021–2023. Digital humanities quarterly (dhq). *Digital Humanities Quarterly*.

Association for Computational Linguistics. 2023. Proceedings of the annual meeting of the association for computational linguistics (acl) 2023. *ACL Anthology*.

Association for Computing Machinery. 2023. Proceedings of the acm on human-computer interaction (cscw), volume 7, issue cscw1. *Proceedings of the ACM on Human-Computer Interaction*.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90, Montreal, Quebec, Canada. Association for Computational Linguistics.

Jari Björne, Filip Ginter, Sampo Pyysalo, Jun'ichi Tsujii, and Tapio Salakoski. 2010. Scaling up biomedical event extraction to the entire PubMed. In *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*, pages 28–36, Uppsala, Sweden. Association for Computational Linguistics.

Claire Bonial, Julia Bonn, Kathryn Conger, Jena D. Hwang, and Martha Palmer. 2014. PropBank: Semantics of new predicate types. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3013–3019, Reykjavik, Iceland. European Language Resources Association (ELRA).

Nancy Chinchor and Beth Sundheim. 1993. MUC-5 evaluation metrics. In *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland, August 25-27, 1993*.

Arman Cohan, Waleed Ammar, Madeleine van Zuylen, and Field Cady. 2019. Structural scaffolds for citation intent classification in scientific publications. 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 3586–3596, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37 – 46.

Carmela Comito, Agostino Forestiero, and Clara Pizzuti. 2019. Bursty event detection in twitter streams. *ACM Trans. Knowl. Discov. Data*, 13(4).

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning.

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. 2004. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Xinya Du and Claire Cardie. 2020a. Document-level event role filler extraction using multi-granularity contextualized encoding. In *Proceedings of the 58th*

Annual Meeting of the Association for Computational Linguistics, pages 8010–8020, Online. Association for Computational Linguistics.

Xinya Du and Claire Cardie. 2020b. Event extraction by answering (almost) natural questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 671–683, Online. Association for Computational Linguistics.

Seth Ebner, Patrick Xia, Ryan Culkin, Kyle Rawlins, and Benjamin Van Durme. 2020. Multi-sentence argument linking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8057–8077, Online. Association for Computational Linguistics.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Sonal Gupta and Christopher Manning. 2011. Analyzing the dynamics of research by extracting key aspects of scientific papers. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1–9, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Kata Gábor, Haïfa Zargayouna, Isabelle Tellier, Davide Buscaldi, and Thierry Charnois. 2016. Unsupervised relation extraction in specialized corpora using sequence mining. In *Proceedings of the XVIth Symposium on Intelligent Data Analysis (IDA 2016)*, pages 237–248, Stockholm, Sweden. Springer.

Song He, Xin Peng, Yihan Cai, Xin Li, Zhiqing Yuan, WenLi Du, and Weimin Yang. 2024. ZSEE: A dataset based on zeolite synthesis event extraction for automated synthesis platform. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1791–1808, Mexico City, Mexico. Association for Computational Linguistics.

Zhenzhen Hong, Logan Ward, Kyle Chard, Ben Blaiszik, and Ian Foster. 2021. Challenges and advances in information extraction from scientific literature: a review. *JOM*, 73(10):3383–3400.

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. DEGREE: A data-efficient generation-based event extraction model. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1890–1908, Seattle, United States. Association for Computational Linguistics.

Kuan-Hao Huang, I-Hung Hsu, Tanmay Parekh, Zhiyu Xie, Zixuan Zhang, Prem Natarajan, Kai-Wei Chang, Nanyun Peng, and Heng Ji. 2024. TextEE: Benchmark, reevaluation, reflections, and future challenges in event extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12804–12825, Bangkok, Thailand. Association for Computational Linguistics.

Kung-Hsiang Huang, Sam Tang, and Nanyun Peng. 2021. Document-level entity-based extraction as template generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5257–5269, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kung-Hsiang Huang, Mu Yang, and Nanyun Peng. 2020. Biomedical event extraction with hierarchical knowledge graphs. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1277–1285, Online. Association for Computational Linguistics.

Tomoki Ikoma and Shigeki Matsubara. 2023. Paper recommendation using citation contexts in scholarly documents. In *Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation*, pages 710–716, Hong Kong, China. Association for Computational Linguistics.

Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. 2020. SciREX: A challenge dataset for document-level information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7506–7516, Online. Association for Computational Linguistics.

JMIR Publications. 2023. Journal of medical internet research (jmir), 2023. *Journal of Medical Internet Research*.

David Jurgens, Srijan Kumar, Raine Hoover, Dan McFarland, and Dan Jurafsky. 2018. Measuring the evolution of a scientific field through citation frames. *Transactions of the Association for Computational Linguistics*, 6:391–406.

Jin-Dong Kim, Yue Wang, Toshihisa Takagi, and Akinori Yonezawa. 2011. Overview of Genia event task in BioNLP shared task 2011. In *Proceedings of BioNLP Shared Task 2011 Workshop*, pages 7–15, Portland, Oregon, USA. Association for Computational Linguistics.

Jin-Dong Kim, Yue Wang, and Yamamoto Yasunori. 2013. The Genia event extraction shared task, 2013 edition - overview. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 8–15, Sofia, Bulgaria. Association for Computational Linguistics.

Suchetha Nambanoor Kunnath, David Pride, Bikash Gyawali, and Petr Knoth. 2020. Overview of the 2020 WOSP 3C citation context classification task. In *Proceedings of the 8th International Workshop on Mining Scientific Publications*, pages 75–83, Wuhan, China. Association for Computational Linguistics.

Alexandria Leto, Shamik Roy, Alexander Hoyle, Daniel Acuna, and Maria Leonor Pacheco. 2024. A first step towards measuring interdisciplinary engagement

in scientific publications: A case study on NLP + CSS research. In *Proceedings of the Sixth Workshop on Natural Language Processing and Computational Social Science (NLP+CSS 2024)*, pages 144–158, Mexico City, Mexico. Association for Computational Linguistics.

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 894–908, Online. Association for Computational Linguistics.

Maria Liakata, Suraj Saha, Simon Dobnik, Colin Batchelor, and Dietrich Rebholz-Schuhmann. 2012. Automatic recognition of conceptualization zones in scientific articles and two life science applications. *Bioinformatics*, 28(7):991–1000.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out (WAS 2004)*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232, Brussels, Belgium. Association for Computational Linguistics.

Yubo Ma, Zehao Wang, Mukai Li, Yixin Cao, Meiqi Chen, Xinze Li, Wenqi Sun, Kunquan Deng, Kun Wang, Aixin Sun, and Jing Shao. 2022. MMEKG: Multi-modal event knowledge graph towards universal representation across modalities. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 231–239, Dublin, Ireland. Association for Computational Linguistics.

Himanshu Maheshwari, Bhavyajeet Singh, and Vasudeva Varma. 2021. SciBERT sentence representation for citation context classification. In *Proceedings of the Second Workshop on Scholarly Document Processing*, pages 130–133, Online. Association for Computational Linguistics.

Meta AI. 2024. Introducing Llama 3.1: Our Most Capable Models to Date.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

MUC, editor. 1992. *Fourth Message Understanding Conference (MUC-4): Proceedings of a Conference Held in McLean, Virginia, June 16-18, 1992*. Association for Computational Linguistics, McLean, Virginia.

Suchetha N. Kunnath, David Pride, Drahomira Herrmannova, and Petr Knoth. 2021. Overview of the 2021 SDP 3C citation context classification shared task. In *Proceedings of the Second Workshop on Scholarly Document Processing*, pages 150–158, Online. Association for Computational Linguistics.

Kiem-Hieu Nguyen, Xavier Tannier, Olivier Ferret, and Romaric Besançon. 2016. A dataset for open event extraction in English. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1939–1943, Portorož, Slovenia. European Language Resources Association (ELRA).

Kei Okamura. 2019. Interdisciplinarity revisited: evidence for research impact and dynamism. *Palgrave Communications*, 5(1):141.

Yvette Oortwijn, Thijs Ossenkoppele, and Arianna Betti. 2021. Interrater disagreement resolution: A systematic procedure to reach consensus in annotation tasks. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 131–141, Online. Association for Computational Linguistics.

OpenAI. 2025. Introducing GPT-4.1 in the API.

Oxford University Press. 2023. Bioinformatics journal. *Bioinformatics*.

Brian Paltridge. 2002. Thesis and dissertation writing: an examination of published advice and actual practice. *English for Specific Purposes*, 21(2):125–143.

Amir Pouran Ben Veyseh, Javid Ebrahimi, Franck Dernoncourt, and Thien Nguyen. 2022. MEE: A novel multilingual event extraction dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9603–9613, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Massimiliano Pronesti, Joao Bettencourt-Silva, Paul Flanagan, Alessandra Pascale, Oisin Redmond, Anya Belz, and Yufang Hou. 2025. Query-driven document-level scientific evidence extraction from biomedical studies.

Sampo Pyysalo, Tomoko Ohta, Makoto Miwa, Han-Cheol Cho, Jun'ichi Tsujii, and Sophia Ananiadou. 2012. Event extraction across multiple levels of biological organization. *Bioinformatics*, 28(18):i575–i581.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. Atomic: an atlas of machine commonsense for if-then reasoning. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA. Association for Computational Linguistics.

Omar Sharif, Joseph Gatto, Madhusudan Basak, and Sarah Masud Preum. 2024. Explicit, implicit, and scattered: Revisiting event extraction to capture complex arguments. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12061–12081, Miami, Florida, USA. Association for Computational Linguistics.

Zhiyi Song, Ann Bies, Stephanie Strassel, Tom Riese, Justin Mott, Joe Ellis, Jonathan Wright, Seth Kulick, Neville Ryant, and Xiaoyi Ma. 2015. From light to rich ERE: Annotation of entities, relations, and events. In *Proceedings of the 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98, Denver, Colorado. Association for Computational Linguistics.

Petros Stavropoulos, Ioannis Lyris, Natalia Manola, Ioanna Grypari, and Haris Papageorgiou. 2023. Empowering knowledge discovery from scientific literature: A novel approach to research artifact analysis. In *Proceedings of the 3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS 2023)*, pages 37–53, Singapore. Association for Computational Linguistics.

MeiHan Tong, Bin Xu, Shuai Wang, Meihuan Han, Yixin Cao, Jiangqi Zhu, Siyu Chen, Lei Hou, and Juanzi Li. 2022. DocEE: A large-scale and fine-grained benchmark for document-level event extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3970–3982, Seattle, United States. Association for Computational Linguistics.

Chen-Tse Tsai, Gourab Kundu, and Dan Roth. 2013. Concept-based analysis of scientific literature. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, CIKM '13, page 1733–1738, New York, NY, USA. Association for Computing Machinery.

U.S. National Library of Medicine. 2023. Structured abstracts.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. Web Download. LDC Catalog No. LDC2006T06.

Yuqing Wang and Yun Zhao. 2024. Metacognitive prompting improves understanding in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1914–1926, Mexico City, Mexico. Association for Computational Linguistics.

Zhongqing Wang and Yue Zhang. 2017. A neural model for joint event detection and summarization. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 4158–4164. AAAI Press.

Hongming Zhang, Xin Liu, Haojie Pan, Yangqiu Song, and Cane Wing-Ki Leung. 2020. Aser: A large-scale eventuality knowledge graph. In *Proceedings of The Web Conference 2020*, WWW '20, page 201–211, New York, NY, USA. Association for Computing Machinery.

Qi Zhang, Zhijia Chen, Huitong Pan, Cornelia Caragea, Longin Jan Latecki, and Eduard Dragut. 2024. SciER: An entity and relation extraction dataset for datasets, methods, and tasks in scientific documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13083–13100, Miami, Florida, USA. Association for Computational Linguistics.

Yuchen Zhuang, Yinghao Li, Junyang Zhang, Yue Yu, Yingjun Mou, Xiang Chen, Le Song, and Chao Zhang. 2022. ReSel: N-ary relation extraction from scientific text and tables by learning to retrieve and select. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 730–744, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Domain-wise Arguments Distribution Analysis

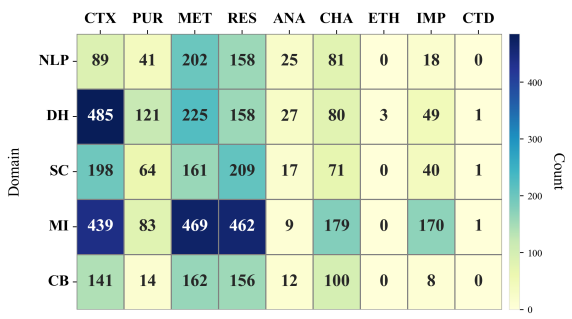

Figure 9: Distribution of argument types across all domains

We report the distribution of argument types across scientific domains (Figure 9). While all domains emphasize Results, Digital Humanities (DH) is a notable exception, being dominated by Context arguments. Among STEM domains—Natural Language Processing (NLP), Computational Biology (CB), and Medical Informatics (MI)—Method arguments are the most prevalent, reflecting their methodological focus. In contrast, DH and Social Computing (SC) place more emphasis on Context and Results, respectively, aligning with the rhetorical nature of these fields. Notably, MI contains the highest number of arguments overall, likely due to the length of its abstracts, even though fewer were annotated to balance domain coverage.

B Codebook details

B.1 Annotation Tool

We deploy our annotation tool on Render[3]. Figure 10 shows our annotation interface.

B.2 Event Type Definition

- **Background:** Briefly outlines the context, motivation, and problem being addressed. It highlights the research gap and the paper's objectives or research questions.
- **Method:** Summarizes the methodologies, frameworks, or techniques used to conduct the study, including experimental setups, algorithms, datasets, or analytical tools.
- **Result:** Reports the main outcomes of the research, emphasizing key data, trends, or discoveries. Focuses on what was achieved or learned.
- **Conclusion:** Discusses the significance of the findings, their impact on the field, potential applications, and how they address the initial problem or research gap. May include recommendations or future research directions.

B.2.1 Trigger Definition

- **Action:** The most representative verb or verb phrase in the event, including auxiliary verbs like *am*, *is*, *are*, *have*, and *has*.
- **Agent:** The entity responsible for initiating or performing the Action, such as a person, system, method, or organization.
- **Object:** The entity that receives, is affected by, or is the focus of the Action (e.g., a concept, result, or entity being acted upon). During annotation, Objects may be separated; in such cases, annotate them as *Primary Object* and *Secondary Object*. Include only the Object spans themselves, and do not include the separators or intervening material.

B.3 Argument Definition

- **Context**
  Definition: Provides foundational or situational information of the event.
  Example: Deep learning has revolutionized **natural language processing tasks**, enabling state-of-the-art results in translation, summarization, and question answering.
- **Purpose**
  Definition: Defines the purpose or aim of the event.
  Example: This study aims to **develop a lightweight transformer model** suitable for deployment on edge devices.
- **Method**
  Definition: Techniques, tools, methodology, or frameworks used in the event.
  Example: We employ **a combination of knowledge distillation and parameter pruning** to reduce model size while maintaining accuracy.
- **Result**
  Definition: Observations or outputs of the event.
  Example: The proposed method achieves a **40% reduction in model size** with only a **1% drop in accuracy on the GLUE benchmark**.
- **Analysis**
  Definition: Interpretation or explanation of other arguments.
  Example: The slight decrease in accuracy can be attributed to **the removal of redundant parameters** that minimally impact overall model performance.
- **Challenge**
  Definition: Constraints or weaknesses of the context, method, or results.
  Example: One significant limitation of the approach is its **dependency on large-scale**

[3] https://render.com/

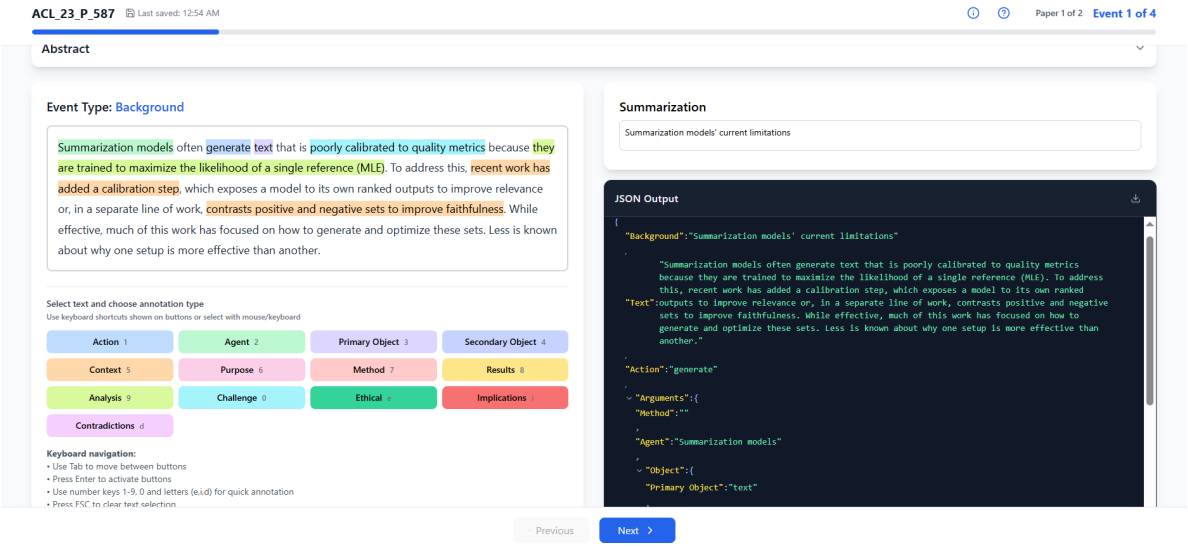

Figure 10: Annotation Tool Interface

labeled datasets for effective knowledge distillation.

- **Ethical**
  Definition: Ethical concerns, implications, and justifications of the event.
  Example: The deployment of these models must address concerns about potential biases in training data that could **unfairly disadvantage certain user groups**.

- **Implication**
  Definition: Broader applicability, significance, or potential for future research.
  Example: Our approach opens the door for **deploying advanced NLP models** on low-power devices, paving the way for **accessible AI in remote or resource-constrained environments**.

- **Contradiction**
  Definition: Disagreements with existing knowledge.
  Example: Contrary to previous studies suggesting that **parameter pruning significantly reduces accuracy**, our results demonstrate **minimal performance loss with careful pruning strategies**.

B.4 Additional Annotation Rules

- **Annotate by Breaking Down Sentences:** Please annotate segments of a sentence (a part of a sentence) instead of a full sentence if different segments of the sentence can be fit into different arguments.

- **Passive Tense:** In a passive tense structure: Something (Agent) + is done (Passive Verb) + by Someone/Something (Object).

- **Indirect Object:** If there is no direct object, you should leave the Object empty.

- **Entire Clause as Object:** In the following structure, the entire clause is the <Object>: <Agent> + <Actions like: show, demonstrate, illustrate, prove, found, explain, indicate, conclude, etc.> + that / what / who / which / where / when / how / whether + clause.

- **Text that Fits Multiple Arguments:** If a text span can fit into multiple <Arguments>, follow this order of importance: **Results** > Purpose > Method > Analysis > Implication > Challenge > Contradiction > Context > Ethical. Results is the most important, and Ethical is the least.

- **Abbreviation:** You should use both the original term and its abbreviation when both are given together, e.g., Chain-of-Thought (CoT), not only Chain-of-Thought or CoT.

- **Use of Primary and Secondary Object:** Annotate as *Primary Object* and *Secondary Object* when an Object is expressed in two separate spans within a sentence. Two common cases occur:

  (1) **Parallel Objects:** The structure is *Action + Primary Object + (and / as well as / also) + Secondary Object*. Intervening clauses (e.g., "which . . . ", "that . . . ") may appear between

Objects; ignore these clauses and annotate only the Object spans. *Example:* "We analyze protein sequences, which exhibit structural variation, and gene expression profiles." Annotation: <Primary Object>: protein sequences; <Secondary Object>: gene expression profiles.

(2) **Transformation Objects:** The structure is *Action + Primary Object + (into / to / for) + Secondary Object*. The *Primary Object* is what is transformed, and the *Secondary Object* is where it is mapped or placed. Always include the preposition introducing the Secondary Object. *Example:* "We aimed to map the most frequently discussed factors into health systems and practical use." Annotation: <Primary Object>: the most frequently discussed factors; <Secondary Object>: into health systems and practical use.

In both cases, please only annotate separate spans of Primary Object and Secondary Object, DO NOT include intervening descriptive clauses or anything in between.

C Prompts

Considering the sensitivity of LLM performance to prompt phrasing, we explore a variety of prompt variations to identify the optimal one. These variations include incorporating the Metacognitive Prompting technique (Wang and Zhao, 2024), adopting role definitions with direct information extraction instead of a QA format, and paraphrasing prompt instructions—for example, replacing "### Output (JSON only)" with "### Your Answer (JSON format)." Each prompt version was evaluated in at least three runs to assess prediction stability, and the final prompt was selected based on its consistent performance across trials.

In this section, we present the prompt designs for each task. We include the Zero-Shot prompt for *Scientific Abstract Segmentation* and *Trigger Identification & Argument Extraction*, and the One-Shot prompt for *Trigger Identification & Argument Extraction*. Additionally, we provide the two event type awareness prompt as well, (1) *True Event-Type Trigger Identification & Argument Extraction* and (2) *Predict Event-Type Trigger Identification & Argument Extraction*.

Zero-Shot Scientific Abstract Segmentation Prompt

You are a strict extraction assistant. Never explain, never repeat, only extract in the required format.
 # # # Abstract: # # #
{abstract}
 # # # Extraction Rules: # # #

- Copy full, continuous sentences from the abstract. No changes, summaries, or guessing allowed.
- Each sentence must belong to only one section.
- Sections must use continuous text spans. No skipping around.
- If no content fits a section, output exactly <NONE>.
- No explanations, no extra text, no format changes.

 # # # Section Definitions: # # #

- **Background**: Problem, motivation, context, research gap, or objectives.
- **Method**: Techniques, experimental setups, frameworks, datasets.
- **Result**: Main findings, discoveries, statistics, or trends.
- **Conclusion**: Importance, impact, applications, or future work.

 # # # Exact Output Format: # # #

```
[Background]: <EXACT TEXT or <NONE>>

[Method]: <EXACT TEXT or <NONE>>

[Result]: <EXACT TEXT or <NONE>>

[Conclusion]: <EXACT TEXT or <NONE>>
```

Zero-Shot Trigger Identification & Argument Extraction Prompt

You are an expert argument annotator. Given a part of a scientific abstract, you need to identify the key trigger for the event (the main verb or action that signals an important research activity) and annotate the abstract with the corresponding argument components related to this trigger. Extractions should capture complete phrases around this key trigger and be organized in a single JSON format, containing only what is explicitly stated in the text without adding any interpretation.

 # # # Abstract Segment to Analyze:

{abstract}

 # # # Argument Components to Extract:

Action: What is the SINGLE most representative trigger (verb or verb phrase) in the segment?
Agent: Who or what is performing the Action?
Object:

- **Primary Object:** What is directly receiving or affected by the Action?
- **Secondary Object:** What is a secondary entity also receiving the Action?

Context: What provides foundational or situational information of the event?
Purpose: What is the purpose or aim of the event?
Method: What techniques, tools, approaches, or frameworks are used in the event?
Results: What are the outcomes, observations or findings of the event?
Analysis: What are the interpretations or explanations of other arguments?
Challenge: What are the constraints or weaknesses of the event?
Ethical: What are the ethical concerns, justifications or implications of the event?
Implications: What is the broader significance or potential for future applications/research?
Contradictions: What are the disagreements with existing knowledge?

 # # # Extraction Rules:

1. Extract complete phrases, not just single words.
2. Only extract elements that are explicitly present. Mark missing elements as ["<NONE>"].
3. Use the exact text from the abstract.
4. Break down sentences when different parts fit different arguments.
5. NEVER use the same span of text for multiple arguments - each piece of text must be assigned to exactly one argument type. However, multiple text spans can be part of the same argument (e.g., ["text span 1", "text span 2".....] can be used for a single argument type) if different parts of the text contribute to the same argument.
6. If text could fit multiple arguments, prioritize in this order: Results > Purpose > Method > Analysis > Implication > Challenge > Contradiction > Context > Ethical

 # # # Output Format:

```
{
 "Action": "EXACT TEXT or <NONE>",
 "Agent": ["EXACT TEXT or <NONE>"],
 "Object": {
  "Primary Object": ["EXACT TEXT or <NONE>"],
  "Secondary Object": ["EXACT TEXT or <NONE>"]
 },
 "Context": ["EXACT TEXT or <NONE>"],
 "Purpose": ["EXACT TEXT or <NONE>"],
 "Method": ["EXACT TEXT or <NONE>"],
 "Results": ["EXACT TEXT or <NONE>"],
 "Analysis": ["EXACT TEXT or <NONE>"],
 "Challenge": ["EXACT TEXT or <NONE>"],
 "Ethical": ["EXACT TEXT or <NONE>"],
 "Implications": ["EXACT TEXT or <NONE>"],
 "Contradictions": ["EXACT TEXT or <NONE>"]
}
```

 # # # IMPORTANT INSTRUCTIONS:

- You MUST return ONLY ONE JSON structure.
- NO explanation text, thinking, or commentary before or after the JSON.
- NEVER repeat the JSON structure.
- ALL fields must use arrays with ["<NONE>"] for missing arguments.
- Follow the EXACT format shown in the template.
- ONLY extract arguments that are explicitly present in the text. DO NOT hallucinate or add any information not found in the abstract.

 # # # Output (JSON only)

One-Shot Trigger Identification & Argument Extraction Prompt

You are an expert argument annotator. Given a part of a scientific abstract, you need to identify the key trigger for the event (the main verb or action that signals an important research activity) and annotate the abstract with the corresponding argument components related to this trigger. Extractions should capture complete phrases around this key trigger and be organized in a single JSON format, containing only what is explicitly stated in the text without adding any interpretation.
 # # # Abstract Segment to Analyze:
{abstract}
 # # # Argument Components to Extract:
Action: What is the SINGLE most representative trigger (verb or verb phrase) in the segment?
Agent: Who or what is performing the Action?
Object:

- **Primary Object:** What is directly receiving or affected by the Action?
- **Secondary Object:** What is a secondary entity also receiving the Action?

Context: What provides foundational or situational information of the event?
Purpose: What is the purpose or aim of the event?
Method: What techniques, tools, approaches, or frameworks are used in the event?
Results: What are the outcomes, observations or findings of the event?
Analysis: What are the interpretations or explanations of other arguments?
Challenge: What are the constraints or weaknesses of the event?
Ethical: What are the ethical concerns, justifications or implications of the event?
Implications: What is the broader significance or potential for future applications/research?
Contradictions: What are the disagreements with existing knowledge?
 # # # Extraction Rules:

1. Extract complete phrases, not just single words.
2. Only extract elements that are explicitly present. Mark missing elements as ["<NONE>"].
3. Use the exact text from the abstract.
4. Break down sentences when different parts fit different arguments.
5. NEVER use the same span of text for multiple arguments - each piece of text must be assigned to exactly one argument type. However, multiple text spans can be part of the same argument (e.g., ["text span 1", "text span 2".....] can be used for a single argument type) if different parts of the text contribute to the same argument.
6. If text could fit multiple arguments, prioritize in this order: Results > Purpose > Method > Analysis > Implication > Challenge > Contradiction > Context > Ethical

Here is a one-shot example of a complete abstract:
Background Event
For abstract: "Second language acquisition (SLA) research has extensively studied cross-linguistic transfer, the influence of linguistic structure of a speaker's native language [L1] on the successful acquisition of a foreign language [L2]. Effects of such transfer can be positive (facilitating acquisition) or negative (impeding acquisition). We find that NLP literature has not given enough attention to the phenomenon of negative transfer."
Output:

```
{
 "Action": "has extensively studied",
 "Agent": ["Second language acquisition (SLA) research"],
 "Object": {
  "Primary Object": ["cross-linguistic transfer"],
  "Secondary Object": ["<NONE>"]
 },
 "Context": ["Effects of such transfer can be positive (facilitating acquisition) or negative (impeding acquisition)"],
 "Purpose": ["<NONE>"],
 "Method": ["<NONE>"],
 "Results": ["<NONE>"],
 "Analysis": ["<NONE>"],
 "Challenge": ["We find that NLP literature has not given enough attention to the phenomenon of negative transfer"],
 "Ethical": ["<NONE>"],
 "Implications": ["<NONE>"],
 "Contradictions": ["<NONE>"]
}
```


Method Event

For abstract: "To understand patterns of both positive and negative transfer between L1 and L2, we model sequential second language acquisition in LMs. Further, we build a Multilingual Age Ordered CHILDES (MAO-CHILDES) — a dataset consisting of 5 typologically diverse languages, i.e., German, French, Polish, Indonesian, and Japanese — to understand the degree to which native Child-Directed Speech (CDS) [L1] can help or conflict with English language acquisition [L2]."
Output:


```
{
 "Action": "model",
 "Agent": ["we"],
 "Object": {
  "Primary Object": ["sequential second language acquisition in LMs"],
  "Secondary Object": ["<NONE>"]
 },
 "Context": ["<NONE>"],
 "Purpose": ["To understand patterns of both positive and negative transfer between L1 and L2"],
 "Method": ["we build a Multilingual Age Ordered CHILDES (MAO-CHILDES)"],
 "Results": ["<NONE>"],
 "Analysis": ["a dataset consisting of 5 typologically diverse languages, i.e., German, French, Polish, Indonesian, and
Japanese"],
 "Challenge": ["<NONE>"],
 "Ethical": ["<NONE>"],
 "Implications": ["<NONE>"],
 "Contradictions": ["<NONE>"]
}
```


Result Event

For abstract: "To examine the impact of native CDS, we use the TILT-based cross lingual transfer learning approach established by Papadimitriou and Jurafsky (2020) and find that, as in human SLA, language family distance predicts more negative transfer. Additionally, we find that conversational speech data shows greater facilitation for language acquisition than scripted speech data."
Output:


```
{
 "Action": "use",
 "Agent": ["we"],
 "Object": {
  "Primary Object": ["the TILT-based cross lingual transfer learning approach"],
  "Secondary Object": ["<NONE>"]
 },
 "Context": ["<NONE>"],
 "Purpose": ["To examine the impact of native CDS"],
 "Method": ["<NONE>"],
 "Results": ["as in human SLA, language family distance predicts more negative transfer", "conversational speech data
shows greater facilitation for language acquisition than scripted speech data"],
 "Analysis": ["<NONE>"],
 "Challenge": ["<NONE>"],
 "Ethical": ["<NONE>"],
 "Implications": ["<NONE>"],
 "Contradictions": ["<NONE>"]
}
```


Conclusion Event

For abstract: "Our findings call for further research using our novel Transformer-based SLA models and we would like to encourage it by releasing our code, data, and models."
Output:

```
{
 "Action": "call for",
 "Agent": ["Our findings"],
 "Object": {
  "Primary Object": ["further research"],
  "Secondary Object": ["<NONE>"]
 },
 "Context": ["<NONE>"],
 "Purpose": ["<NONE>"],
 "Method": ["using our novel Transformer-based SLA models"],
 "Results": ["<NONE>"],
 "Analysis": ["<NONE>"],
 "Challenge": ["<NONE>"],
 "Ethical": ["<NONE>"],
 "Implications": ["we would like to encourage it by releasing our code, data, and models"],
 "Contradictions": ["<NONE>"]
}
```

Output Format:

```
{
 "Action": "EXACT TEXT or <NONE>",
 "Agent": ["EXACT TEXT or <NONE>"],
  "Object": {
  "Primary Object": ["EXACT TEXT or <NONE>"],
  "Secondary Object": ["EXACT TEXT or <NONE>"]
 },
 "Context": ["EXACT TEXT or <NONE>"],
 "Purpose": ["EXACT TEXT or <NONE>"],
 "Method": ["EXACT TEXT or <NONE>"],
 "Results": ["EXACT TEXT or <NONE>"],
 "Analysis": ["EXACT TEXT or <NONE>"],
 "Challenge": ["EXACT TEXT or <NONE>"],
 "Ethical": ["EXACT TEXT or <NONE>"],
 "Implications": ["EXACT TEXT or <NONE>"],
 "Contradictions": ["EXACT TEXT or <NONE>"]
}
```

IMPORTANT INSTRUCTIONS:

- You MUST return ONLY ONE JSON structure.
- NO explanation text, thinking, or commentary before or after the JSON.
- NEVER repeat the JSON structure.
- ALL fields must use arrays with ["<NONE>"] for missing arguments.
- Follow the EXACT format shown in the template.
- ONLY extract arguments that are explicitly present in the text. DO NOT hallucinate or add any information not found in the abstract.
- Carefully study the one-shot examples to understand how arguments should be correctly annotated from the text.

Output (JSON only)

True Event-Type Trigger Identification & Argument Extraction Prompt

You are an expert argument annotator. Given a part of the text and the event type from the scientific abstract (e.g., "Background", "Method", "Result", "Conclusion"), you need to identify the key trigger for the event (the main verb or action that signals an important research activity) and annotate the abstract with the corresponding argument components related to this trigger. Extractions should capture complete phrases around this key trigger and be organized in a single JSON format, containing only what is explicitly stated in the text without adding any interpretation.
 # # # Event Type Definitions:

- **Background**: Problem, motivation, context, research gap, or objectives.
- **Method**: Techniques, experimental setups, frameworks, datasets.
- **Result**: Main findings, discoveries, statistics, or trends.
- **Conclusion**: Importance, impact, applications, or future work.

 # # # {event_type} Event Abstract Segment to Analyze: # # #
{abstract}
 # # # Argument Components to Extract:
Action: What is the SINGLE most representative trigger (verb or verb phrase) in the segment?
Agent: Who or what is performing the Action?
Object:

- **Primary Object:** What is directly receiving or affected by the Action?
- **Secondary Object:** What is a secondary entity also receiving the Action?

Context: What provides foundational or situational information of the event?
Purpose: What is the purpose or aim of the event?
Method: What techniques, tools, approaches, or frameworks are used in the event?
Results: What are the outcomes, observations or findings of the event?
Analysis: What are the interpretations or explanations of other arguments?
Challenge: What are the constraints or weaknesses of the event?
Ethical: What are the ethical concerns, justifications or implications of the event?
Implications: What is the broader significance or potential for future applications/research?
Contradictions: What are the disagreements with existing knowledge?
 # # # Extraction Rules:

1. Extract complete phrases, not just single words.
2. Only extract elements that are explicitly present. Mark missing elements as ["<NONE>"].
3. Use the exact text from the abstract.
4. Break down sentences when different parts fit different arguments.
5. NEVER use the same span of text for multiple arguments - each piece of text must be assigned to exactly one argument type. However, multiple text spans can be part of the same argument (e.g., ["text span 1", "text span 2".....] can be used for a single argument type) if different parts of the text contribute to the same argument.
6. If text could fit multiple arguments, prioritize in this order: Results > Purpose > Method > Analysis > Implication > Challenge > Contradiction > Context > Ethical
7. Consider the event type when determining the most appropriate argument assignments.

Output Format:

```
{
 "Action": "EXACT TEXT or <NONE>",
 "Agent": ["EXACT TEXT or <NONE>"],
  "Object": {
  "Primary Object": ["EXACT TEXT or <NONE>"],
  "Secondary Object": ["EXACT TEXT or <NONE>"]
 },
 "Context": ["EXACT TEXT or <NONE>"],
 "Purpose": ["EXACT TEXT or <NONE>"],
 "Method": ["EXACT TEXT or <NONE>"],
 "Results": ["EXACT TEXT or <NONE>"],
 "Analysis": ["EXACT TEXT or <NONE>"],
 "Challenge": ["EXACT TEXT or <NONE>"],
 "Ethical": ["EXACT TEXT or <NONE>"],
 "Implications": ["EXACT TEXT or <NONE>"],
 "Contradictions": ["EXACT TEXT or <NONE>"]
}
```

IMPORTANT INSTRUCTIONS:

- You MUST return ONLY ONE JSON structure.
- NO explanation text, thinking, or commentary before or after the JSON.
- NEVER repeat the JSON structure.
- ALL fields must use arrays with ["<NONE>"] for missing arguments.
- Follow the EXACT format shown in the template.
- ONLY extract arguments that are explicitly present in the text. DO NOT hallucinate or add any information not found in the abstract.
- Use the provided event type to guide your analysis and ensure the extraction is appropriate for that type of event.

Output (JSON only)

Predict Event-Type Trigger Identification & Argument Extraction Prompt

You are an expert argument annotator. Given a part of the text from a scientific abstract, you need to first determine what type of event this text represents, then identify the key trigger for the event (the main verb or action that signals an important research activity) and annotate the abstract with the corresponding argument components related to this trigger. Based on the event type you determine, perform the argument extraction accordingly. Extractions should capture complete phrases around this key trigger and be organized in a single JSON format, containing only what is explicitly stated in the text without adding any interpretation.

 # # # Event Type Definitions:

- **Background**: Problem, motivation, context, research gap, or objectives.
- **Method**: Techniques, experimental setups, frameworks, datasets.
- **Result**: Main findings, discoveries, statistics, or trends.
- **Conclusion**: Importance, impact, applications, or future work.

 # # # Abstract Segment to Analyze: # # #
{abstract}
 # # # Argument Components to Extract:
Action: What is the SINGLE most representative trigger (verb or verb phrase) in the segment?
Agent: Who or what is performing the Action?
Object:

- **Primary Object:** What is directly receiving or affected by the Action?
- **Secondary Object:** What is a secondary entity also receiving the Action?

Context: What provides foundational or situational information of the event?
Purpose: What is the purpose or aim of the event?
Method: What techniques, tools, approaches, or frameworks are used in the event?
Results: What are the outcomes, observations or findings of the event?
Analysis: What are the interpretations or explanations of other arguments?
Challenge: What are the constraints or weaknesses of the event?
Ethical: What are the ethical concerns, justifications or implications of the event?
Implications: What is the broader significance or potential for future applications/research?
Contradictions: What are the disagreements with existing knowledge?
 # # # Extraction Rules:

1. Extract complete phrases, not just single words.
2. Only extract elements that are explicitly present. Mark missing elements as ["<NONE>"].
3. Use the exact text from the abstract.
4. Break down sentences when different parts fit different arguments.
5. NEVER use the same span of text for multiple arguments - each piece of text must be assigned to exactly one argument type. However, multiple text spans can be part of the same argument (e.g., ["text span 1", "text span 2".....] can be used for a single argument type) if different parts of the text contribute to the same argument.
6. If text could fit multiple arguments, prioritize in this order: Results > Purpose > Method > Analysis > Implication > Challenge > Contradiction > Context > Ethical

7. Consider the event type when determining the most appropriate argument assignments.

 # # # Output Format:

```
{
 "Action": "EXACT TEXT or <NONE>",
 "Agent": ["EXACT TEXT or <NONE>"],
  "Object": {
  "Primary Object": ["EXACT TEXT or <NONE>"],
  "Secondary Object": ["EXACT TEXT or <NONE>"]
 },
 "Context": ["EXACT TEXT or <NONE>"],
 "Purpose": ["EXACT TEXT or <NONE>"],
 "Method": ["EXACT TEXT or <NONE>"],
 "Results": ["EXACT TEXT or <NONE>"],
 "Analysis": ["EXACT TEXT or <NONE>"],
 "Challenge": ["EXACT TEXT or <NONE>"],
 "Ethical": ["EXACT TEXT or <NONE>"],
 "Implications": ["EXACT TEXT or <NONE>"],
 "Contradictions": ["EXACT TEXT or <NONE>"]
}
```

 # # # IMPORTANT INSTRUCTIONS:

- You MUST return ONLY ONE JSON structure.
- NO explanation text, thinking, or commentary before or after the JSON.
- NEVER repeat the JSON structure.
- ALL fields must use arrays with ["<NONE>"] for missing arguments.
- Follow the EXACT format shown in the template.
- ONLY extract arguments that are explicitly present in the text. DO NOT hallucinate or add any information not found in the abstract.
- Use the provided event type to guide your analysis and ensure the extraction is appropriate for that type of event.

 # # # Output (JSON only)

Methods	Arg-I (EM)			Arg-C (EM)		
	P	R	F1	P	R	F1
Tuning-based models						
EEQA	14.26	15.01	14.63	11.59	12.20	11.88
DEGREE	**44.97**	12.69	19.79	**34.23**	9.66	15.07
OneIE	32.03	**35.27**	**33.57**	25.38	**27.95**	**26.61**
Zero-shot LLMs						
DS-R1-Llama	10.33	5.46	7.15	6.23	3.30	4.31
Qwen	9.59	7.10	8.16	5.08	3.76	4.33
Llama	7.01	7.17	7.09	3.73	3.81	3.77
GPT	17.84	23.03	20.10	13.37	17.27	15.07
One-shot LLMs						
DS-R1-Llama	13.28	5.51	7.79	7.08	2.93	4.15
Qwen	13.98	9.16	11.07	7.24	4.74	5.73
Llama	13.02	9.93	11.27	6.55	5.00	5.67
GPT	25.75	25.79	25.77	19.38	19.41	19.4

Table 6: EM-based Precision (P), Recall (R), and F1-score (%) on baseline models for argument identification (Arg-I) and classification (Arg-C) tasks.

D Argument Extraction with EM Metrics and detailed Human Performance Comparison

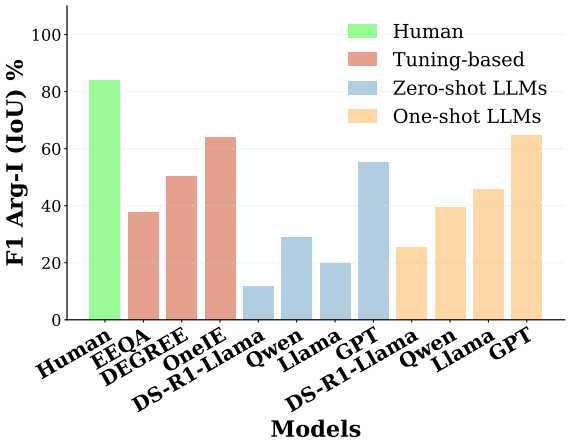

Figure 11: Performance comparison of various methods on argument identification (Arg-I) using IoU F1 scores. Methods are grouped by type: Human baseline, tuning-based models, zero-shot LLMs, and one-shot LLMs.

In Section 6, we analyze argument extraction under the IoU metric and examined the human–model performance gap for argument classification (Arg-C) using IoU. Here, we complement that analysis by reporting results under the EM metric for argument extraction, as well as argument identification (Arg-I) and trigger identification with ROUGE-L human–model gaps, to provide a more comprehensive evaluation. As shown in Table 6, OneIE remains the best-performing model, while DEGREE continues to exhibit high precision but low recall. Among LLMs, GPT-4.1 consistently achieves the

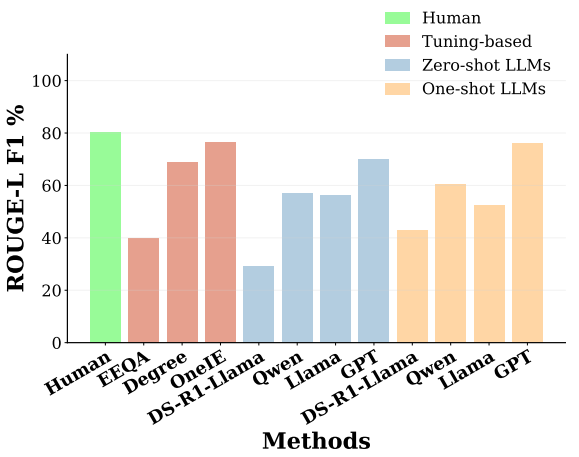

Figure 12: Performance comparison of various methods on ROUGE-L F1 scores. Methods are grouped by type: Human baseline, tuning-based models, zero-shot LLMs, and one-shot LLMs.

best performance, and one-shot prompting again improves results across all LLMs. Overall, the findings remain consistent—switching from IoU to EM does not alter the relative comparison between models, but EM results in lower scores for all models due to its stricter matching criteria.

Figure 11 shows the Arg-I performance gap between humans and models, which closely mirrors the Arg-C results. The gap remains around 20%, highlighting the need for multi-domain scientific EE models. In contrast, Figure 12 reveals a smaller gap in ROUGE-L scores for trigger identification, indicating that this task is considerably easier and most models perform well. Nevertheless, since argument extraction is the core challenge, there remains significant room for improvement in addressing multi-domain scientific EE.

E Effects of removal of domains on each tuning-based model

We present domain ablation results for DEGREE and EEQA under the EM setting in Figure 13 and Figure 14, respectively. For DEGREE, removing a domain consistently leads to performance drops, similar to OneIE, though the impact is generally smaller. This suggests DEGREE benefits from domain-specific training but is somewhat more resilient, possibly due to its generative nature. In contrast, EEQA shows minimal sensitivity to domain removal. This may be because its QA-based design relies more on question formulation and span selection, making it less dependent on domain-specific linguistic patterns.

F Trigger and Argument Identification by Event Types and Domains

Results for trigger identification and argument identification are presented by event type and domain, providing supplementary detail to the analysis in Section 6 and offering deeper insight into how event types and domains impact SciEvent performance.

Figure 15 presents ROUGE-L scores for trigger identification by event type, where the Conclusion event achieves the highest performance. This is likely due to its shorter and simpler structure, offering fewer candidate verbs, making trigger extraction easier. The performance trends for other event types are similar to those discussed in Sections 6 on argument classification. Figure 16 reports IoU scores for argument identification, which closely mirror the argument classification results but show an overall performance increase of about 20%, due to the easier argument identification task.

Figure 17 shows some difference in the Medical Informatics (MI) domain compared to argument classification. MI exhibits lower trigger identification performance, due to longer texts containing more verbs, which increases ambiguity and makes trigger extraction more difficult. Figure 18 again shows a 20% performance boost across all models, due to the easier argument identification task, while preserving trends consistent with those observed in argument classification.

G SciEvent keywords analysis

We present a detailed keyword analysis grounded in each domain's call for papers in table 7. For Digital Humanities (DHq 2021—2023), we include the majority of abstracts from 2021 to 2023 due to limited publications, ensuring comprehensive coverage and minimizing bias. On the other hand, as shown in the tables below for the rest four domains, we observe that our dataset covers all major research topics outlined in each venue's call for papers. This suggests that our benchmark includes a diverse set of scientific articles and is reasonably representative within each domain.

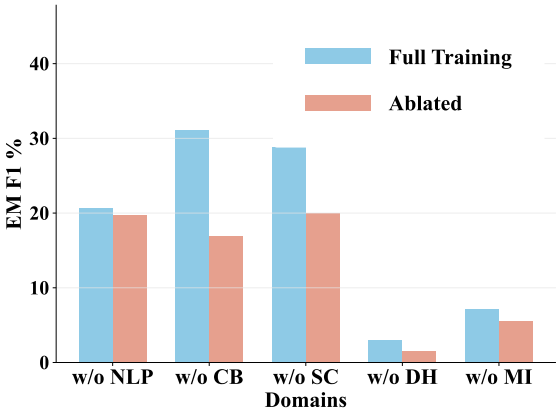

Figure 13: F1-scores reported for full training versus training with one domain removed for DEGREE under Exact Match (EM).

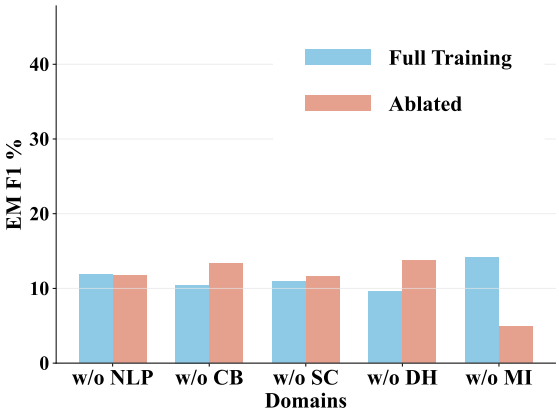

Figure 14: F1-scores reported for full training versus training with one domain removed for EEQA under Exact Match (EM).

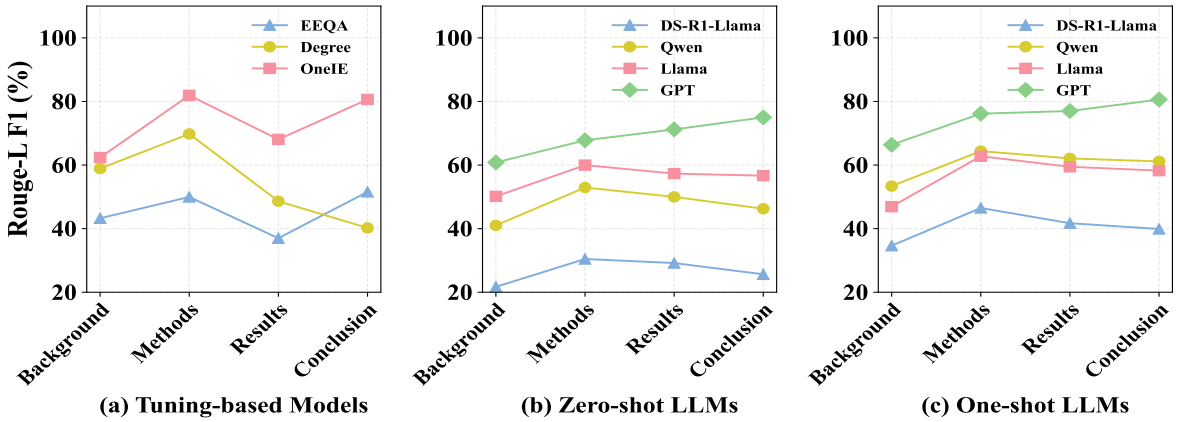

Figure 15: Comparison of Rouge-L F1 scores (%) across different event types.

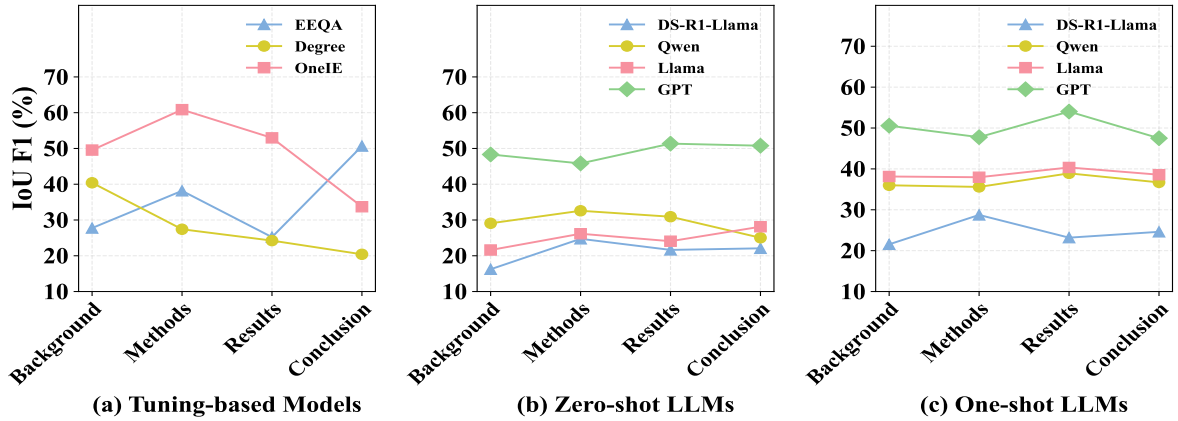

Figure 16: Comparison of Intersection-over-Union (IoU) on Arg-I F1-scores (%) across different event types.

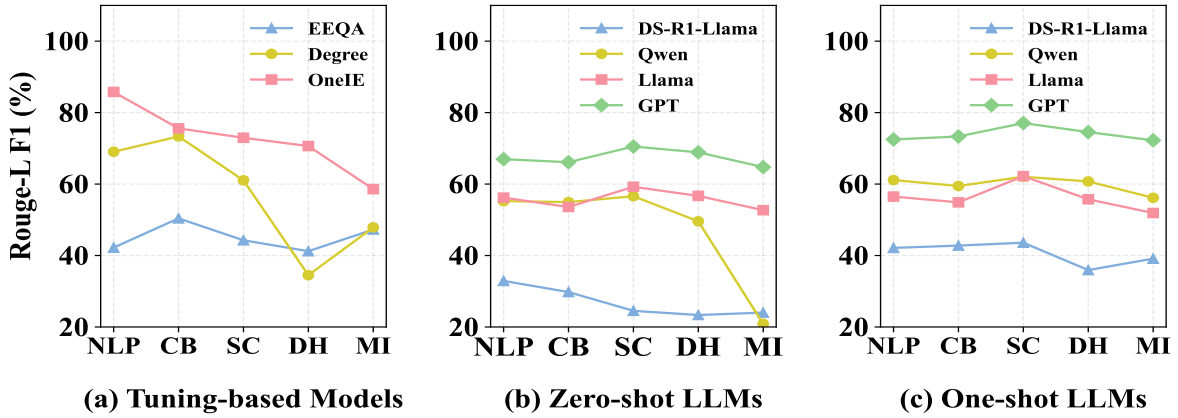

Figure 17: Comparison of Rouge-L F1 scores (%) across different academic domains.

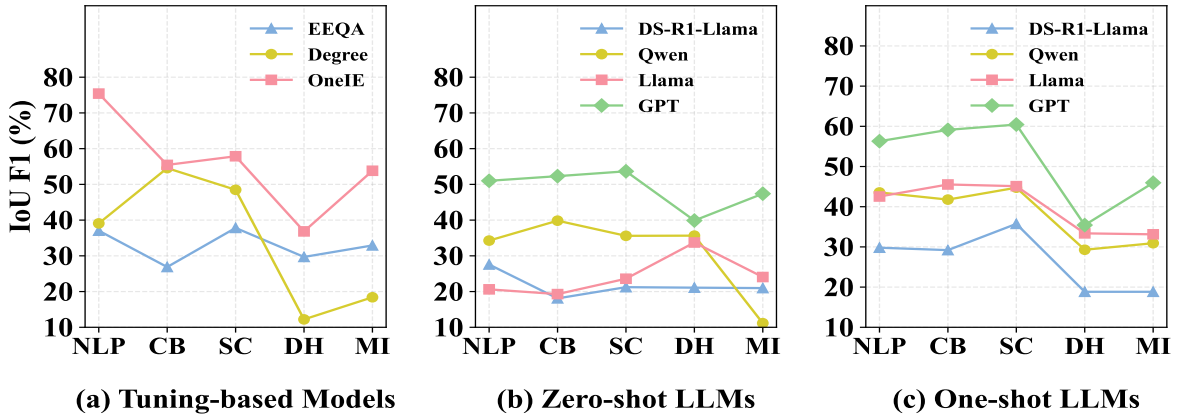

Figure 18: Comparison of Intersection-over-Union (IoU) on Arg-I F1-scores (%) across different academic domains.

H Detailed example of SciEvent dataset

We show one detailed example of SciEvent dataset, including event segmentation and event extraction in Figrue 19

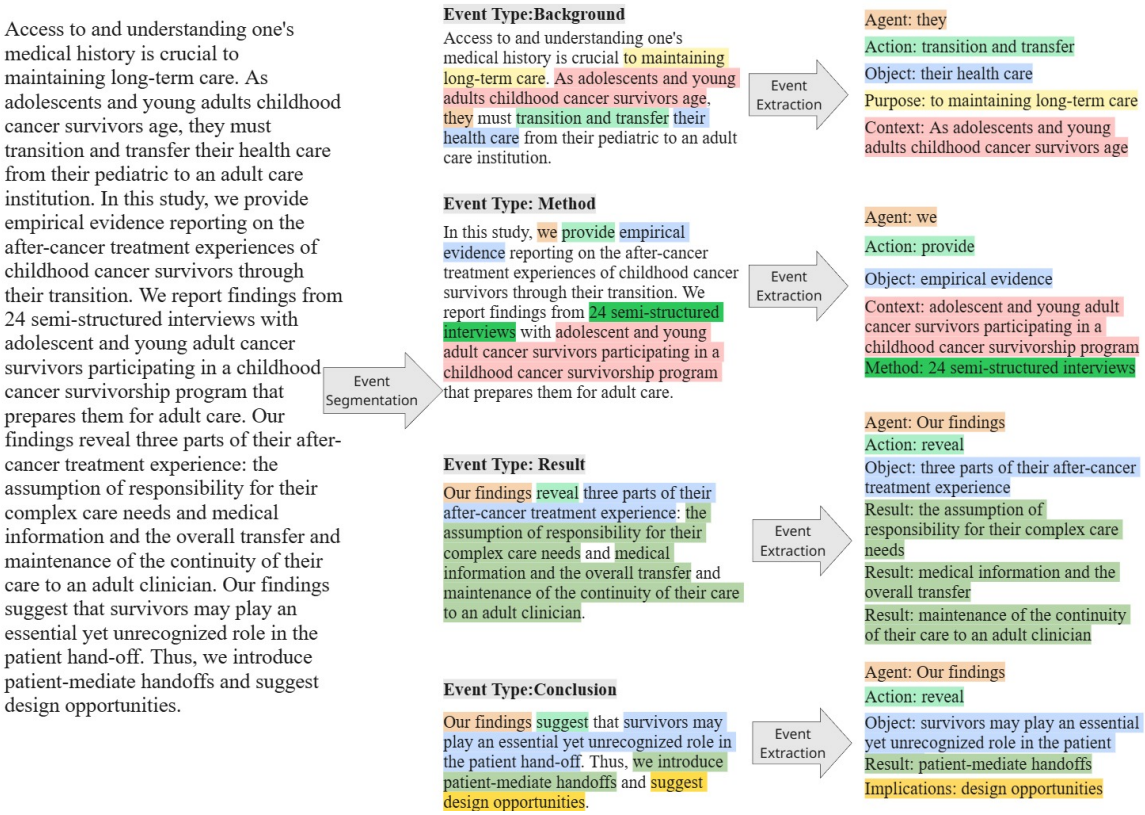

Figure 19: Full event extraction example from SciEvent, including event segmentation and event extraction, where trigger is a tuple including Agent, Action and Object.

Domain	Count
NLP (ACL 2023)	
Computational Social Science and Cultural Analytics	2
Dialogue and Interactive Systems	10
Discourse and Pragmatics	2
Ethics and NLP	8
Generation	26
Information Extraction	10
Information Retrieval and Text Mining	2
Interpretability and Analysis of Models for NLP	24
Language Grounding to Vision, Robotics and Beyond	2
Multilingualism and Language Contact	16
Linguistic Theories, Cognitive Modeling, and Psycholinguistics	6
Machine Learning for NLP	**50**
Machine Translation	6
NLP Applications	3
Phonology, Morphology, and Word Segmentation	2
Question Answering	8
Resources and Evaluation	**62**
Semantics: Lexical	2
Semantics: Sentence-level, Textual Inference, Other Areas	4
Sentiment, Stylistic, Argument Mining	6
Speech and Multimodality	10
Summarization	6
Syntax: Tagging, Chunking and Parsing	6
CB (Bioinformatics 2023)	
Genome analysis	6
Sequence analysis	4
Phylogenetics	4
Structural bioinformatics	14
Gene expression	16
Genetic and population analysis	**18**
Systems biology	14
Data and text mining	6
Databases and ontologies	12
Bioimage informatics	4
SC (CSCW 2023)	
Social and crowd computing	67
System development	6
Theory	42
Empirical investigations	**78**
Data mining and modeling	27
Methodologies and tools	**77**
Domain-specific social and collaborative applications	31
Collaboration systems based on emerging technologies	7
Ethics and policy implications	33
Crossing boundaries	19
MI (JMIR 2023)	
Clinical Decision Support	**47**
Automated Feedback	7
Virtual Patient Development	25
Content Quality and Prompting	23
AI Curriculum Design	11
Patient Education via ChatGPT	19
Preparing for AI-Literate Patients	9
Ethics and Legal Concerns	**44**
Academic Integrity and Policy	15
Trends and Use Cases	24
Future Outlook	3
Practical Tutorials	8

Table 7: Domain distribution and counts across different research venues and conferences.