SciER: An Entity and Relation Extraction Dataset for Datasets, Methods, and Tasks in Scientific Documents

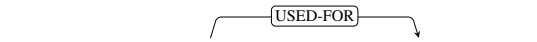
Qi Zhang[1] **Zhijia Chen**[1] **Huitong Pan**[1]
Cornelia Caragea[2] **Longin Jan Latecki**[1] **Eduard Dragut**[1]
[1]Temple University [2]University of Illinois Chicago
{qi.zhang, latecki, edragut}@temple.edu, cornelia@uic.edu

Abstract

Scientific information extraction (SciIE) is critical for converting unstructured knowledge from scholarly articles into structured data (entities and relations). Several datasets have been proposed for training and validating SciIE models. However, due to the high complexity and cost of annotating scientific texts, those datasets restrict their annotations to specific parts of paper, such as abstracts, resulting in the loss of diverse entity mentions and relations in context. In this paper, we release a new entity and relation extraction dataset for entities related to datasets, methods, and tasks in scientific articles. Our dataset contains 106 *manually annotated* full-text scientific publications with over 24k entities and 12k relations. To capture the intricate use and interactions among entities in full texts, our dataset contains a fine-grained tag set for relations. Additionally, we provide an out-of-distribution test set to offer a more realistic evaluation. We conduct comprehensive experiments, including state-of-the-art supervised models and our proposed LLM baselines, and highlight the challenges presented by our dataset, encouraging the development of innovative models to further the field of SciIE.[1]

1 Introduction

Scientific Information Extraction (SciIE) is a core topic of scientific literature mining (Luan et al., 2017; Groth et al., 2018; Sadat and Caragea, 2022; Park and Caragea, 2023; Pan et al., 2024a). It typically includes scientific named entity extraction (SciNER) and scientific relation extraction (SciRE), and plays a critical role in downstream applications, including scientific knowledge graph construction (Wang et al., 2021; Gautam et al., 2023), data searching (Viswanathan et al., 2023), academic question answering (Dasigi et al., 2021), and method recommendation (Luan et al., 2018). Scientific large language models (LLMs) like Galactica (Taylor et al., 2022) enable several practical applications such as citations suggestion, scientific question answering (QA), and scientific code generation (Li et al., 2023). However, their generated content is frequency-biased, often exhibits overconfidence, and lacks factual basis (Xu et al., 2023). SciIE, integrated with suitable retrieval, and QA systems can mitigate those issues and enhance model effectiveness in downstream tasks (Shu et al., 2022; Xu et al., 2023).



	E_1: METHOD	E_2: TASK
Task	Input	Output
NER	S_1	E_1, E_2
RE	S_1, [E_1, E_2]	USED-FOR
ERE	S_1	[E_1, USED-FOR, E_2]

Figure 1: Top: An annotation sample of our SciER dataset, illustrating the labeling process and data structure. The sentence S_1 contains two annotated spans denoting two entities E_1 and E_2, with respective types METHOD and TASK. Bottom: A table detailing the input and output of the three tasks supported by our SciER dataset, including Named Entity Recognition (NER), Relation Extraction (RE), and Entity and Relation Extraction (ERE).

SciIE faces unique challenges compared to general domain IE. First, data annotation for SciIE is highly dependent on expert annotators, resulting in a scarcity of high-quality labeled datasets. Second, SciIE needs to handle more complex text, which evolves constantly with novel terminology, unlike general domain IE. For instance, SciIE faces more severe temporal and conceptual shifts (Zhang et al., 2019; Viswanathan et al., 2021; Zaporojets et al., 2022; Chen et al., 2022, 2024; Pham et al., 2023), whereas fundamental entities and relationships in general IE tend to remain more static over time compared to those in the scientific literature.

[1]Dataset and code are publicly available: https://github.com/edzq/SciER

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 13083–13100
November 12-16, 2024

Existing SciIE datasets and benchmarks that support both SciNER and SciRE are limited to extracting information from specific parts of papers, such as particular paragraphs (Augenstein et al., 2017) or abstracts (Gábor et al., 2018; Luan et al., 2018). However, scientific entities like datasets, methods, and tasks entities, are distributed throughout the entire text of papers. Sentences in the body of a paper exhibit diverse linguistic styles and ways to mention entities (Li et al., 2023) and semantics (Jain et al., 2020), which allows the extraction of more fine-grained and precise relation types. For example, abstracts do not say that method X is trained on dataset Y, but experimental sections give such details. Therefore, focusing on specific parts of scientific articles is likely to miss important information. Several datasets (Pan et al., 2024b, 2023; Otto et al., 2023; Jain et al., 2020) attempt to create SciIE benchmarks with full-text annotation, but they ignore the SciRE task.

In this paper, we present SciER, an entity and relation extraction dataset for identifying dataset, method, and task entities in scientific documents as well as the relations between them. Our dataset is large, with 24K entities and 12k relations from 106 scientific articles, enabling the evaluation and development of SciIE models. These documents are taken from the publications included in Papers with Code (PwC)[2], covering artificial intelligence (AI) topics, such as natural language processing (NLP), machine learning (ML), computer vision (CV), and AI for Science (AI4Science). Figure 1 shows an annotated sentence from our dataset, which gives the entities, their types, i.e., `METHOD` and `TASK`, respectively, and the relation between them `USED-FOR`. Our dataset can be used to evaluate NER and RE as separate tasks, but it can also support the evaluation of end-to-end entity and relation extraction (ERE) from scientific publications (Luan et al., 2018; Ye et al., 2022). The table in Figure 1 describes those settings. For example, in NER the input is a sentence and the output is the set of entities in the sentence. In RE, the input is the sentence along with the entities and the output is the relation between those entities. Finally, in ERE the triplet <*subject*, *relation*, *object*> is the expected output from a sentence.

We address the limitations of existing datasets by annotating entire scientific papers for both entity and their relations. This is a much harder task compared to annotating abstracts. Furthermore, comparing with existing datasets (Augenstein et al., 2017; Gábor et al., 2018; Luan et al., 2018), we provide more fine-grained relation types to describe the interactions between datasets, methods, and tasks. For example, we use `TRAINED-WITH` and `EVALUATED-WITH` to describe the interactions between methods and datasets. These relation types need to be extracted from the body of a paper, and are not supported by previous datasets. §3.3 gives a detailed comparison between our dataset and existing ones. Finally, to evaluate the model's robustness to temporal and conceptual shifts in the SciIE, we set in-distributed (ID) and out-of-distribution (OOD) test sets. The documents in the OOD set were all published after the training documents and feature entirely different topics. We conduct evaluation experiments by employing three state-of-the-art supervised methods and LLMs-based in-context learning (ICL) methods and provide analysis. Specifically, for LLMs-based methods, we tested both pipeline and joint approaches, optimizing the prompts through retrieval-based ICL, tag-based entity extraction, and the incorporation of annotation guidelines. The experimental results show that for LLMs, pipeline modeling, which splits the ERE task into two sub-tasks of NER and RE, outperformas joint extraction. In the challenging ERE task, the best supervised method achieves an F1 score of 61.10%, while the best LLM method achieves an F1 score of 41.22%.

Our contributions can be summarized as follows:

- We provide a manually annotated dataset consisting of 106 full-text scientific publications, containing over 24k entities and 12k relations. Our dataset is significantly larger than previous datasets that support both SciNER and SciRE tasks.
- We introduce a fine-grained tag set designed for scientific relation extraction, customized to reflect the use and interaction of machine learning datasets, methods, and tasks entities in scientific publications.
- We conducted experiments on LLMs baselines using both pipeline and joint approaches. We optimized the prompt through retrieval-based ICL, tag-based entity extraction, and the incorporation of annotation guidelines. We also provided a comparative analysis between LLMs methods and three state-of-the-art supervised baselines, highlighting the key challenges.

[2] https://paperswithcode.com/

	SemEva17	SemEval18	SciERC	SciER
Annotation Unit	♣	♦	♦	♠
#Entity Types	3	-	6	3
#Relation Types	2	6	7	9
#Entities	9946	7483	8089	24518
#Relations	672	1595	4716	12083
#Docs	500	500	500	106
#Relations/Doc	1.3	3.2	9.4	114.0

Table 1: Comparison of SciER and 3 datasets supporting NER and RE in scientific text. Annotation units: ♣=Paragraph, ♦=Abstract, ♠=Full Text.

2 Related Work

Many datasets for SciNER have been proposed. (Heddes et al., 2021) and DMDD (Pan et al., 2023) are two datasets for dataset mention detection. The (Heddes et al., 2021) dataset comprises 6000 annotated sentences selected based on the occurrence of dataset related word patterns from four major AI conference publications. DMDD is annotated on the full text and comprises 31219 scientific articles automatically annotated with distant supervision (Zhang et al., 2018). TDMSci (Hou et al., 2021) supports three types of entities: TASK, DATASET, and METHOD. It has 2000 sentences extracted from NLP papers. SciREX (Jain et al., 2020) offers comprehensive coverage with 438 full text annotated documents and supports four entity types: TASK, DATASET, METHOD, and METRIC. SciREX does not annotate relations between pairs of those entity types. (Otto et al., 2023) manually annotates 100 documents for fine-grained SciNER by defining 10 different entity types in 3 categories: MLModel related, Dataset related and miscellaneous. SciDMT (Pan et al., 2024b) uses the PwC as knowledge created a very large scale dataset for DATA, METHOD, and TASK. SciDMT includes 48 thousand scientific articles with over 1.8 million weakly annotated mention annotations in their main corpus. However, given the inherent complexity of the NER task, employing weak labels may cause models to overfit on noisy data, thereby substantially impacting their performance (Liu et al., 2021; Bhowmick et al., 2022, 2023).

Although there has been growing interest in research on developing methods and datasets for SciIE, very few datasets support both NER and RE tasks for scientific text. An overview of existing SciIE benchmarks that support both SciNER and SciRE is shown in Table 1. SEMEVAL-2017 TASK 10 (SemEval 17) (Augenstein et al., 2017) includes 500 paragraphs from open-access journals and supports three types of entities: TASK, METHOD, and MATERIAL and two relation types: HYPONYM-OF and SYNONYM-OF. SEMEVAL-2018 TASK 7 (SemEval 18) (Gábor et al., 2018) has been proposed for predicting six types of relations between entities. All sentences in SemEval 18 are from the abstracts of NLP papers and have only entity spans (i.e., without annotation of entity types). SciERC (Luan et al., 2018) contains 500 scientific abstracts with the annotations for scientific entities, their relations, and coreference clusters. SciERC defines six types of entities and seven types of relations. However, these three datasets are limited on annotating abstracts or pre-selected paragraphs. Thus, a significant number of sentences that contain more diverse entity mention forms and semantics are lost.

Compared to those resources, our dataset contains 106 scientific publications with minute manual annotations. The dataset has nine relation types, allowing for more nuanced relations between entities. The scale of our dataset, which contains more than 24k entities and over 12k relations, which is significantly larger than previous datasets, except for those that are created with distant supervision.

3 SciER

In this section, we detail the curation of our dataset, including data collection process in §3.1, the data annotation process in §3.2, and present the final dataset statistics and comparisons in §3.3.

3.1 Data Collection and Processing

Our dataset includes 106 documents from two sources. ❶ One hundred of these documents come from the SciDMT validation set (SciDMT-E). These documents are from the PwC website and we use the corresponding PDF parsed version released by the S2ORC (Lo et al., 2020). These papers cover different machine learning topics and have publication dates prior to 2022. We re-check the entity annotations from SciDMT-E[3] and then add relation annotations. ❷ We selected additional six papers from top AI conferences as an out-of-distribution (OOD) test set. To simulate a more realistic application scenario, we chose these six papers published in 2023-2024, four of which focus on AI4Science topics not included in the first 100 documents. For these six OOD test documents, we first collected their PDF files and then used Grobid (GRO, 2008–2024) for parsing.

[3]We provide details of our re-checking workload on SciDMT-E in the Appendix A.1.

3.2 Data Annotation

Annotation Scheme For the entity annotation, we use the SciDMT annotation scheme, which defined three types of entities: DATASET, METHOD, and TASK. To maintain consistency with the PwC website database, we only annotate the factual entities, unlike previous works (Luan et al., 2018; Otto et al., 2023) which annotate both factual and non-factual entities. For example, the "CoNLL03" and "SNLI" are factual entities, but the "a high-coverage sense-annotated corpus" is not a factual entity.

For the relation annotation, we define nine fine-grained tag set to establish interaction relationships between datasets, methods, and tasks entities in scientific documents. They are EVALUATED-WITH, COMPARE-WITH, SUBCLASS-OF, BENCHMARK-FOR, TRAINED-WITH, USED-FOR, SUBTASK-OF, PART-OF, and SYNONYM-OF. Directionality is taken into account except for the two symmetric relation types (SYNONYM-OF and COMPARE-WITH). We provide our semantic relation typology and corresponding examples in Table 2. Specifically, compared to previous datasets (Augenstein et al., 2017; Luan et al., 2018; Gábor et al., 2018), we employ more specific relation types for identical entity types and extend usage relations among different types of entities in a more granular manner. For example, we use SUBTASK-OF and SUBCLASS-OF to describe the hierarchical relations between tasks and methods, respectively. This can provide better interpretability and allows for direct usage in practical applications such as building taxonomies. Additionally, we use TRAINED-WITH and EVALUATED-WITH to describe the more precise interactions between methods and datasets. We provide more detailed definitions of the labels for entities and relations in our annotation guidelines in Appendix E.

Annotation Strategy We have five annotators with backgrounds in computer science and machine learning. We conduct the annotation using INCEpTION[4] platform. All annotators had annotation training before starting to annotate on assigned documents. For the 100 documents from SciDMT-E, we asked annotators to first re-check the SciNER annotation before proceeding to the SciRE annotation. For the six OOD documents, annotators need to annotate both SciNER and SciRE from scratch.

Human Agreement One annotator leads the entire annotation process and annotates all the documents in the dataset and each document is also annotated by at least two other annotators. For the first 100 documents, the kappa score (Davies and Fleiss, 1982) for entity annotation is 94.2%, relation annotation is 70.8%; for the six OOD documents, the kappa score for entity annotation is 74.1%, relation annotation is 73.8%. The almost perfect agreement of entity annotation on the first 100 documents is because we derive the original annotation from SciDMT-E.

[4] https://inception-project.github.io/

3.3 Dataset Statistics and Comparison

After the annotation process, our dataset contains over 24k entities and 12k relations, with each document averaging about 114 relations. As shown in Table 1, our dataset is significantly larger than previous datasets supporting both entity and relation extraction task. Specifically, for the widely used SciERC dataset, when we only consider Dataset, Method, and Task entities, it contains only about 1.5k entity and 1.5k relation annotations, where more details are provided in Appendix A.2. We randomly split the first 100 documents into train, development, and ID test sets, containing 80, 10, and 10 documents, respectively. We used six OOD documents as the OOD test set. Appendix A.3 lists the number of samples for each relation type in each set of our dataset.

4 Experiments

In this section, we provide the details of evaluation experiments of both state-of-the-art supervised baselines and LLMs-based baselines on the proposed dataset. We first formally define the problem of end-to-end relation extraction in §4.1, then describe the supervised methods in §4.2 and the LLMs-based methods in §4.3. Finally, we present our implementation details in §4.4 and evaluation settings in §4.5.

4.1 Problem Definition

We aim our dataset as a means to train and evaluate SciIE models. Formally, the input document is denoted as D, which contains a sequence of paragraphs $P = \{p_1, p_2, ..., p_n\}$. Each paragraph p is composed of a sequence of sentences $\{s_1, s_2, ..., s_n\}$ and each sentence is composed of a sequence of words $\{w_1, w_2, ..., w_n\}$. Formally, the problem of end-to-end relation extraction can be decomposed into two sub-tasks:

Named Entity Recognition Let $\mathbb{E}$ denote a set of pre-defined entity types. The NER task is to

Relation Type	Explanation	Example
EVALUATED-WITH	Methods are evaluated by datasets	We use COCO to evaluate ConerNet-Lite and compare it wither other detectors.
COMPARE-WITH	Entities are linked by comparison relation	MAC ...outperforms all tested RANSAC-fashion estimators , such as SAC-COT ...
SUBCLASS-OF	One method is a specialized class of another	
BENCHMARK-FOR	Datasets are used to evaluate tasks	FlyingChairs is a synthetic dataset designed for training CNNs to estimate optical flow .
TRAINED-WITH	Methods are trained by datasets	
USED-FOR	Entities are linked by usage relation	
SUBTASK-OF	A specific part of another broader Task	...is critical for dense prediction tasks such as object detection ...
PART-OF	Entities are in a part-whole relation	Adding attention to our deep learning-based network translated to...
SYNONYM-OF	Entities have same or very similar meanings	...to improve Generative Adversarial Network (GAN) for ...

Table 2: Semantic relation typology for DATASET, METHOD, and TASK entities.

identify all entity mentions from the input sentence $s = \{w_1, w_2, ..., w_n\}$. For each identified entity, we need to give its span $e_i = \{w_l, ..., w_r\}$, where l and r represent the left and right word indices of the span, and classify its entity type $t \in \mathbb{E}$.

Relation Extraction Let $\mathbb{R}$ denote a set of predefined relation types. The task is to predict the relation type $r \in \mathbb{R}$ for every pair of entities (e_i, e_j), if one exists, and $r = \{\text{NULL}\}$ otherwise. Since end-to-end relation extraction comprises two subtasks, this task is typically addressed using ❶ joint entity and relation extraction (ERE) or ❷ pipeline extraction, i.e., performing the NER task first and then using the NER results for RE.

4.2 Supervised Baselines

We apply three supervised methods: ❶ **PURE** (Zhong and Chen, 2021) utilizes two independent encoders to perform pipeline extraction. The outputs of entity encoder are fed into the relation encoder to facilitate end-to-end relation extraction. This method emphasizes the significance of unique representations for entities and relations, the early integration of entity information, and leveraging global context to improve performance. ❷ **PL-Marker** (Ye et al., 2022) introduces a novel span representation technique that augments the outputs of pre-trained encoders to perform pipeline extraction. It leverages two specialized packing strategies—neighborhood-oriented for identifying entity boundaries, and subject-oriented for classifying complex span pairs—which helps understand the interrelations between spans. ❸ **HGERE** (Yan et al., 2023) proposes a joint ERE method by incorporating a high-recall pruner to reduce error propagation and by employing a hypergraph neural network to model complex interactions among entities and relations. This approach has led to significant performance improvements, establishing new state-of-the-art results in the joint ERE.

4.3 LLMs-based Baselines

LLMs via in-context learning (ICL) represents a significant advancement in NLP (Qin et al., 2023). To comprehensively evaluate the LLMs' capability on SciIE, we employ LLMs with zero-shot and few-shot settings to perform both pipeline extraction and joint ERE. Several studies suggest that choosing few-shot in-context examples for each test example dynamically instead of using a fixed set of in-context examples yields strong improvements for ICL (Jimenez Gutierrez et al., 2022; Liu et al., 2022). In our experiments, we follow this setting by employing a retriever to find top similar samples from training set as in-context examples. We will first detail the prompt template construction to formalize the NER, RE, and joint ERE as a language generation task (Jimenez Gutierrez et al., 2022). Then we will introduce the specific settings and efforts to improve the prompt for each task.

We construct an unique prompt for each given test example, which is fed to the LLM. Each prompt consists of the following components:

Instruction I The task instruction I provides the LLMs with a basic description of the task the LLM needs to perform and in what format it should output the results.

Demonstrations D The demonstrations are retrieved from the training set for as in-context examples to help the model better understand the task. Specifically, we will employ a retriever to compute the sentence similarity score and acquire the most similar k demonstrations (x_i, y_i) to build D.

Test Input x_{test} Following the same format as demonstrations, we offer the test input x_{test}, and LLM is expected to generate the corresponding output result y_{test}.

In summary, LLMs-based few-shot in-context learning (ICL) for each task can be formulated as:

$$P(y_{test}|I, D, x_{test}) \quad (1)$$

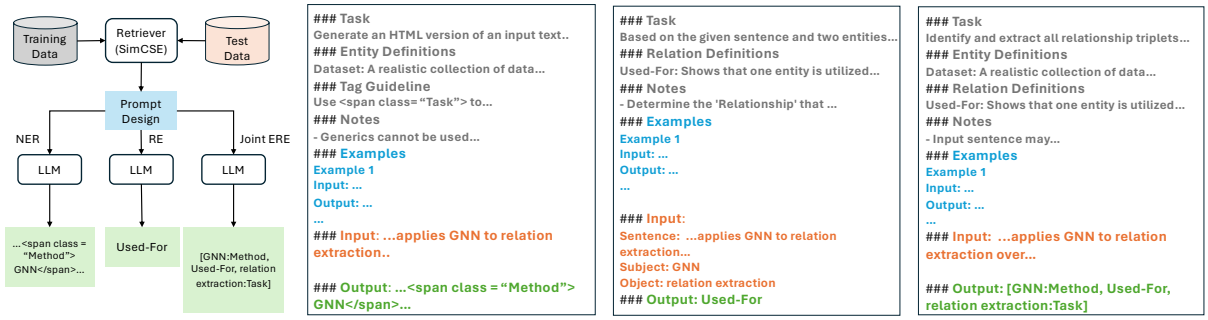


Figure 2: Overall architecture of LLM in-context learning (few-shot) baselines for NER, RE and joint Entity and Relation Extraction (ERE) (first). The few-shot prompt templates for NER (second), RE (third), and Joint ERE (fourth). Different colors indicate different prompt design elements: gray for annotation guideline-based task instructions I, blue for retrieved demonstrations D, orange denotes the test example input x_{test}, and the green represents the expected output of test example output, which will be omitted during testing. y_{test}. Due to space constraints, we shortened the text of our prompts.

When performing zero-shot ICL, D will be removed from the prompt.

Our LLMs-based baseline framework is shown in Figure 2. Existing work indicates that for information extraction tasks, LLMs require clearer instructions to improve the performance(Qin et al., 2023; Hu et al., 2024; Sainz et al., 2023; Jimenez Gutierrez et al., 2022). Therefore, we use annotation guidelines to optimize our prompts. Specifically, for each task, we include two additional instruction components: ❶ **label definitions** and ❷ **annotation notes**. For label definition, we provide definitions of all entities for the NER task, and definitions of all relations for the RE task. For the Joint ERE task, which requires the model to perform both NER and RE simultaneously, we provide definitions of both entities and relations. For annotation notes, we derive suitable instructions from the human annotation guidelines (see Appendix E) for each task and provide them to the LLMs. We believe that introducing entity and relation definitions and annotation notes offers comprehensive and unambiguous descriptions of the target extraction information.

In terms of formatting the NER annotation in prompt, we present it as HTML span tag. This is because (Wadhwa et al., 2023; Hu et al., 2024) demonstrated that when using LLMs for information extraction, the generated results might have the same meaning as in the input text but differs in surface form. For example, the entity "CNNs" in the input sentence might be generated as "CNN". To mitigate this error in NER, we instruct the LLMs use HTML span tags to mark all entities in the input sentence to extract the entity spans and use the class attribute to determine the entity types. For example, the entity "CNNs" in the input text will be marked as "CNNs". We provide the complete prompt used in our experiments in Appendix D.

4.4 Implementation Details

For the supervised methods, we use the *scibert-scivocab-uncased* (Beltagy et al., 2019) as encoder. For the LLMs-based methods, we test the GPT-3.5-Turbo Llama3-70b, and Qwen2-72b as the LLM. For few-shot ICL setting, we retrieve 30 demonstrations for each task, and we use the SimCSE (Gao et al., 2021) as the retriever. For consistent comparison, all experiments are conducted at the sentence-level. Appendix B has additional implementation details.

4.5 Evaluation Settings

To evaluate the pipeline extraction and joint ERE, we compute the performance for each subtask, including NER, end-to-end RE (using NER results for relation extraction), and RE (relation extraction with given gold standard entities). For NER, we conduct span-level evaluation, where both the entity boundary and entity type need to be correctly extracted. For the end-to-end RE, similar to (Zhong and Chen, 2021; Ye et al., 2022; Yan et al., 2023), we report two evaluation metrics: ❶ **Boundaries evaluation** (Rel), which requires the model to correctly predict the boundaries of the subject entity and the object entity, as well as the entity relation; ❷ **Strict evaluation** (Rel+), which further requires the model to predict the entity types based on the requirements of the boundary prediction. For the RE, given any pair of subject and object entity, the model needs to determine whether a pre-defined relation exists. If a relation does exist, the model must predict the corresponding type.

5 Experimental Results

5.1 Main Results

Table 3 reports the experimental results on ID test set and OOD test set. As described in §4.5, for the pipeline extraction methods, we present additional RE results when gold standard entities are given.

Supervised Baselines We observe that HGERE achieves the best performance on both ID and OOD test sets in NER, Rel, and Rel+, demonstrating the robustness of this current SOTA method. When comparing the results of ID and OOD, we find that all methods show performance drop on the OOD test set for NER, Rel, and Rel+. This is because OOD test provides more challenging and realistic validation scenarios, which require the models to extract information from newly published papers containing new entities. We also observe that the decline in NER scores is more significant, especially for PURE and PL-Marker, whose performance dropped by nearly 10 F1 points. This indicates that extracting unseen entities is more challenging for supervised models compared to relation extraction, which is further supported by the slight decline in RE performance for PURE and PL-Marker in OOD compared to ID. We provide a qualitative example in Appendix C.1.

LLMs-based Baselines From the results of both zero-shot and few-shot setting, we have the following observations: ❶ Qwen2-72b exhibits the best overall performance than GPT-3.5-turbo and Llama3-70b in both zero-shot and few-shot settings (except the NER task). ❷ Pipeline extraction outperforms joint ERE in both zero-shot and few-shot settings. Surprisingly, for both Llama3-70b and Qwen2-72b, pipeline extraction shows a significant improvement over joint ERE. We observed that the NER performance in the pipeline extraction is significantly better than in the joint ERE. This indicates that performing LLMs for this end-to-end relation extraction task by decomposing it into seperate NER and RE processes yields better results than joint extraction. ❸ For LLMs-based baselines, the performance of ID does not always outperform OOD and such pattern is very different from supervised baselines. We believe this is due to the extensive training of LLMs on large-scale data. Specifically, for the RE, even though few-shot settings provide similar demonstrations of test data, the ID results are still worse than OOD. However, for the NER, Rel, and Rel+ tasks under few-shot settings, the performance on ID tends to be better than on OOD. Additionally, compared to OOD, the overall performance improvement on ID after using few-shot settings is generally greater than on OOD. This is because, the demonstrations provided to the LLMs are more similar to the ID data.

Previous works (Wan et al., 2023; Jimenez Gutierrez et al., 2022; Ma et al., 2023) showed that information extraction tasks are very challenging for LLMs compared to supervised methods. However, for NER, we found that with appropriate prompt settings, LLMs can be a competent NER model, as reaching an F1 score of 61.69 in zero-shot setting, comparing to the best. This suggests that incorporating LLMs into the NER dataset creation process is a feasible solution to reduce human labor. LLMs perform worse on RE tasks. This is because the test samples for RE tasks contain a large number of NULL labels (see C.2), and large language models have a strong tendency to classify the NULL into predefined types, which has also been confirmed by recent works (Jimenez Gutierrez et al., 2022; Wan et al., 2023). Our experiments show that for end-to-end relation extraction (Rel and Rel+), including the current state-of-the-art (SOTA) models and LLMs-based baselines, there is still significant room for improvement in the future.

5.2 Ablation Study

To validate the effectiveness of the annotation guideline-enhanced prompt design used in LLM-based baselines, we conducted an ablation study using the Llama3-70b model in a few-shot setting. Specifically, for all tasks, we removed the additional instructions derived from the annotation guidelines, retaining only the basic task description in the instruction I. For the NER task, we further removed the requirement of using HTML span tags, allowing the model to directly generate all entities from the input text rather than tagging the input text. Figure 3 presents the results of our ablation study. The results indicate that incorporating label definitions and comprehensive annotation task guidelines significantly improve the model's performance across all tasks. Additionally, for NER, the use of HTML span tags further enhances performance.

5.3 Train Size Experiment

Annotating datasets for information extraction within specific domains presents certain challenges. Comparing to partial text, such as sentence and abstracts, full-text annotation further exacerbates the difficulties for annotation. In the training stage,

Methods	ID Test				OOD Test			
	NER	Rel	Rel+	RE	NER	Rel	Rel+	RE
Supervised Baselines								
PURE (Zhong and Chen, 2021)	81.60	53.27	52.67	73.99	71.99	50.44	49.46	73.63
PL-Marker (Ye et al., 2022)	83.31	60.06	59.24	**77.11**	73.93	59.02	56.68	**76.83**
HGERE (Yan et al., 2023)	**86.85**	**62.32**	**61.10**	-	**81.32**	**61.31**	**58.32**	-
Zero-Shot LLMs-based Baselines								
GPT3.5-Turbo (Joint)	34.76	11.38	10.34	-	37.48	10.95	9.97	-
GPT3.5-Turbo (Pipeline)	51.19	13.57	13.57	35.48	37.73	12.06	11.34	40.74
Llama3-70b (Joint)	48.87	17.31	17.01	-	44.28	17.12	16.63	-
Llama3-70b (Pipeline)	**61.69**	22.28	21.71	37.35	53.09	27.87	25.57	53.87
Qwen2-72b (Joint)	42.15	16.27	14.99	-	40.47	15.54	14.31	-
Qwen2-72b (Pipeline)	58.57	**25.76**	**25.76**	**53.50**	**56.43**	**31.25**	**28.13**	**55.37**
Few-Shot LLMs-based Baselines								
GPT3.5-Turbo (Joint)	62.36	23.71	23.49	-	51.12	20.12	20.12	-
GPT3.5-Turbo (Pipeline)	66.27	27.27	24.94	43.26	55.82	22.37	21.49	44.12
Llama3-70b (Joint)	63.23	29.21	29.16	-	53.12	20.06	19.93	-
Llama3-70b (Pipeline)	**76.02**	37.55	36.74	56.06	**63.98**	31.33	29.64	62.71
Qwen2-72b (Joint)	63.73	35.84	34.87	-	49.21	33.17	33.17	-
Qwen2-72b (Pipeline)	71.44	**41.51**	**41.22**	**60.21**	61.72	**39.12**	**37.13**	**63.93**

Table 3: Test F1 scores of different baselines on our proposed dataset. "Joint" denotes joint ERE. "Pipeline" refers to performing NER and RE separately. "Rel" and "Rel+" denote the results of end-to-end relation extraction under boundaries evaluation and strict evaluation, respectively. "RE" indicates performing relation extraction with given gold standard entities, applicable only to pipeline extraction methods.

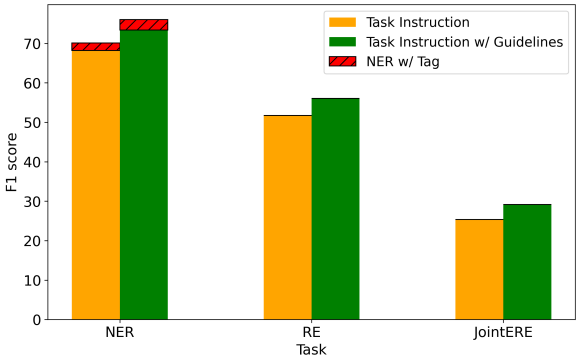


Figure 3: Ablation study for the effectiveness of using annotation guideline to improve the Instruction I. "NER w/ Tag" denotes the performance gain with addtional HTML tag setting.

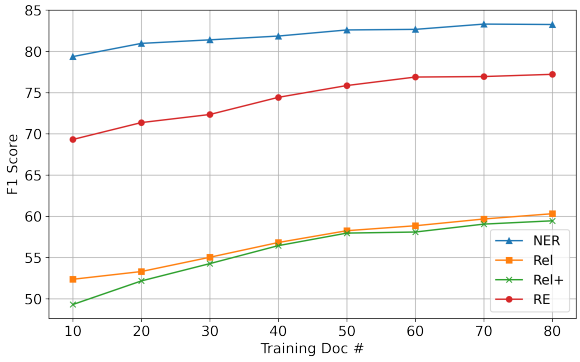


Figure 4: Performance trends of PL-Marker trained on varying number of documents for NER, end-to-end RE (Rel and Rel+) and RE.

the number of fully annotated documents plays a crucial role, as documents with fewer annotations have a significant cost advantage. We conducted an experiment aimed at assessing the performance of different scientific information extraction tasks across different numbers of training documents. Figure 4 shows the performance trends of the training pipeline extraction model PL-Marker for NER, end-to-end RE (Rel and Rel+), and RE. We observe that NER shows a relatively slowed-down improvement as the dataset size increases, suggesting that while it benefits from more data, it experiences diminishing returns when the amount of data becomes large. In contrast, both end-to-end RE (Rel and Rel+) and RE show better improvements with an increase in the number of training documents. This indicates that relation extraction is more data-sensitive, requiring more nuanced and varied annotation data for optimal performance.

6 Conclusion

We introduce SciER, a dataset for entity and relation extraction in scientific documents, specifically focusing on datasets, methods, and task entities. To address the limitations of existing datasets, we annotate entire scientific papers for both entities and relations, resulting in a large-scale dataset comprising 106 full-text scientific publications from various AI topics, containing over 24,000 entities

and 12,000 relations. Additionally, we introduce a fine-grained relation set to describe the interactions between datasets, methods, and tasks. To evaluate the model's robustness to emporal and conceptual shifts in the SciIE, we also set an OOD test set.

We conduct comprehensive evaluation experiments, including supervised state-of-the-art (SOTA) models and LLM-based ICL baselines, to highlight the challenges in this task. Specifically, for LLM-based methods, we tested both pipeline and joint approaches, optimizing the prompts through retrieval-based ICL, tag-based entity extraction, and the incorporation of annotation guidelines. The experimental results of LLMs-based methods show that: ❶ For the ERE task, pipeline modeling, which decomposes the task into NER and RE sub-tasks, significantly outperforms joint modeling; ❷ Although LLM-based approaches require less labeled data, there remains a performance gap compared to supervised methods. For future work, we aim to further optimize prompts to enhance the performance of LLMs in Scientific Information Extraction (SciIE) and domain-specific IE tasks. Additionally, a LLM-in-the-loop data annotation system to reduce the high costs of creating domain-specific IE datasets is feasible.

Limitations

Despite our diligent efforts, developing a gold standard dataset for entity and relation extraction using a fine-grained and comprehensive relation tag set focused on machine learning datasets, methods, and tasks remains a nontrivial undertaking. This leads to the following limitations associated with the creation of our corpus. Our dataset only supports three entity types: `DATASET`, `METHOD`, and `TASK`. Incorporating more diverse entity types would be more beneficial for the development of SciIE. Additionally, many scientific entities are nested, which we have not included. We also observed that parsing documents from PDF format contains some errors, which increases the difficulty of document processing and cause some of our sentences contain errors. Finally, we believe that further evluation experiments can be conducted, such as optimizing the ICL baselines for LLMs. However, due to space constraints, we will consider these as future work.

Ethical Statement

The data included in our newly proposed dataset includes a subset of the data collected and freely published by (Pan et al., 2024b) within the SciDMT project. All the other data are public from scientific documents. We release dataset for scientific information extraction tasks. There are no risks in our work.

Acknowledgements

This work was supported by the National Science Foundation awards III-2107213, III-2107518, and ITE-2333789. We also thank Saiyun Dong and Faezeh Rajabi Kouchi at Temple University, and Seyedeh Fatemeh Ahmadi at UIC for their valuable contributions to our project.

References

2008–2024. Grobid. https://github.com/kermitt2/grobid.

Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. SemEval 2017 task 10: ScienceIE - extracting keyphrases and relations from scientific publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 546–555, Vancouver, Canada. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Satadisha Saha Bhowmick, Eduard C Dragut, and Weiyi Meng. 2022. Boosting entity mention detection for targetted twitter streams with global contextual embeddings. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 1085–1097. IEEE.

Satadisha Saha Bhowmick, Eduard C Dragut, and Weiyi Meng. 2023. Globally aware contextual embeddings for named entity recognition in social media streams. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, pages 1544–1557. IEEE.

Zhijia Chen, Lihong He, Arjun Mukherjee, and Eduard C Dragut. 2024. Comquest: Large scale user comment crawling and integration. In *SIGMOD Conference Companion*, pages 432–435.

Zhijia Chen, Weiyi Meng, and Eduard Dragut. 2022. Web record extraction with invariants. *Proceedings of the VLDB Endowment*, 16(4):959–972.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. 2021. A dataset

of information-seeking questions and answers anchored in research papers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4599–4610, Online. Association for Computational Linguistics.

Mark Davies and Joseph L Fleiss. 1982. Measuring agreement for multinomial data. *Biometrics*, pages 1047–1051.

Kata Gábor, Davide Buscaldi, Anne-Kathrin Schumann, Behrang QasemiZadeh, Haïfa Zargayouna, and Thierry Charnois. 2018. SemEval-2018 task 7: Semantic relation extraction and classification in scientific papers. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 679–688, New Orleans, Louisiana. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nikita Gautam, David Shumway, Megan Kowalcyk, Sarthak Khanal, Doina Caragea, Cornelia Caragea, Hande Mcginty, and Samuel Dorevitch. 2023. Leveraging existing literature on the web and deep neural models to build a knowledge graph focused on water quality and health risks. In *Proceedings of the ACM Web Conference 2023*, pages 4161–4171.

Paul Groth, Mike Lauruhn, Antony Scerri, and Ron Daniel Jr. 2018. Open information extraction on scientific text: An evaluation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3414–3423, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jenny Heddes, Pim Meerdink, Miguel Pieters, and Maarten Marx. 2021. The automatic detection of dataset names in scientific articles. *Data*, 6(8):84.

Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly. 2021. TDMSci: A specialized corpus for scientific literature entity tagging of tasks datasets and metrics. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 707–714, Online. Association for Computational Linguistics.

Yan Hu, Qingyu Chen, Jingcheng Du, Xueqing Peng, Vipina Kuttichi Keloth, Xu Zuo, Yujia Zhou, Zehan Li, Xiaoqian Jiang, Zhiyong Lu, et al. 2024. Improving large language models for clinical named entity recognition via prompt engineering. *Journal of the American Medical Informatics Association*, page ocad259.

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.

Bernal Jimenez Gutierrez, Nikolas McNeal, Clayton Washington, You Chen, Lang Li, Huan Sun, and Yu Su. 2022. Thinking about GPT-3 in-context learning for biomedical IE? think again. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4497–4512, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yuhan Li, Jian Wu, Zhiwei Yu, Börje F Karlsso, Wei Shen, Manabu Okumura, and Chin-Yew Lin. 2023. Unlocking science: Novel dataset and benchmark for cross-modality scientific information extraction. *arXiv preprint arXiv:2311.08189*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Kun Liu, Yao Fu, Chuanqi Tan, Mosha Chen, Ningyu Zhang, Songfang Huang, and Sheng Gao. 2021. Noisy-labeled NER with confidence estimation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3437–3445, Online. Association for Computational Linguistics.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, 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.

Yi Luan, Mari Ostendorf, and Hannaneh Hajishirzi. 2017. Scientific information extraction with semi-supervised neural tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2641–2651, Copenhagen, Denmark. Association for Computational Linguistics.

Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun. 2023. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10572–10601, Singapore. Association for Computational Linguistics.

Wolfgang Otto, Matthäus Zloch, Lu Gan, Saurav Karmakar, and Stefan Dietze. 2023. GSAP-NER: A novel task, corpus, and baseline for scholarly entity extraction focused on machine learning models and datasets. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8166–8176, Singapore. Association for Computational Linguistics.

Huitong Pan, Qi Zhang, Cornelia Caragea, Eduard Dragut, and Longin Jan Latecki. 2024a. Flowlearn: Evaluating large vision-language models on flowchart understanding. *arXiv preprint arXiv:2407.05183*.

Huitong Pan, Qi Zhang, Cornelia Caragea, Eduard Dragut, and Longin Jan Latecki. 2024b. SciDMT: A large-scale corpus for detecting scientific mentions. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14407–14417, Torino, Italia. ELRA and ICCL.

Huitong Pan, Qi Zhang, Eduard Dragut, Cornelia Caragea, and Longin Jan Latecki. 2023. DMDD: A large-scale dataset for dataset mentions detection. *Transactions of the Association for Computational Linguistics*, 11:1132–1146.

Seo Park and Cornelia Caragea. 2023. Multi-task knowledge distillation with embedding constraints for scholarly keyphrase boundary classification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13026–13042, Singapore. Association for Computational Linguistics.

Dong Pham, Xanh Ho, Quang Thuy Ha, and Akiko Aizawa. 2023. Solving label variation in scientific information extraction via multi-task learning. In *Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation*, pages 243–256, Hong Kong, China. Association for Computational Linguistics.

Behrang QasemiZadeh and Anne-Kathrin Schumann. 2016. The ACL RD-TEC 2.0: A language resource for evaluating term extraction and entity recognition methods. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1862–1868, Portorož, Slovenia. European Language Resources Association (ELRA).

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is ChatGPT a general-purpose natural language processing task solver? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1339–1384, Singapore. Association for Computational Linguistics.

Mobashir Sadat and Cornelia Caragea. 2022. SciNLI: A corpus for natural language inference on scientific text. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7399–7409, Dublin, Ireland. Association for Computational Linguistics.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2023. Gollie: Annotation guidelines improve zero-shot information-extraction. In *The Twelfth International Conference on Learning Representations*.

Yiheng Shu, Zhiwei Yu, Yuhan Li, Börje Karlsson, Tingting Ma, Yuzhong Qu, and Chin-Yew Lin. 2022. TIARA: Multi-grained retrieval for robust question answering over large knowledge base. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8108–8121, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*.

Vijay Viswanathan, Luyu Gao, Tongshuang Wu, Pengfei Liu, and Graham Neubig. 2023. DataFinder: Scientific dataset recommendation from natural language descriptions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10288–10303, Toronto, Canada. Association for Computational Linguistics.

Vijay Viswanathan, Graham Neubig, and Pengfei Liu. 2021. CitationIE: Leveraging the citation graph for scientific information extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 719–731, Online. Association for Computational Linguistics.

Somin Wadhwa, Silvio Amir, and Byron Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada. Association for Computational Linguistics.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. GPT-RE: In-context learning for relation extraction using large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547, Singapore. Association for Computational Linguistics.

Qingyun Wang, Manling Li, Xuan Wang, Nikolaus Parulian, Guangxing Han, Jiawei Ma, Jingxuan Tu, Ying Lin, Ranran Haoran Zhang, Weili Liu, Aabhas Chauhan, Yingjun Guan, Bangzheng Li, Ruisong Li, Xiangchen Song, Yi Fung, Heng Ji, Jiawei Han, Shih-Fu Chang, James Pustejovsky, Jasmine Rah, David Liem, Ahmed ELsayed, Martha Palmer, Clare Voss, Cynthia Schneider, and Boyan Onyshkevych. 2021. COVID-19 literature knowledge graph construction and drug repurposing report generation. In

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations, pages 66–77, Online. Association for Computational Linguistics.

Silei Xu, Shicheng Liu, Theo Culhane, Elizaveta Pertseva, Meng-Hsi Wu, Sina Semnani, and Monica Lam. 2023. Fine-tuned LLMs know more, hallucinate less with few-shot sequence-to-sequence semantic parsing over Wikidata. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5778–5791, Singapore. Association for Computational Linguistics.

Zhaohui Yan, Songlin Yang, Wei Liu, and Kewei Tu. 2023. Joint entity and relation extraction with span pruning and hypergraph neural networks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7512–7526, Singapore. Association for Computational Linguistics.

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun. 2022. Packed levitated marker for entity and relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4904–4917, Dublin, Ireland. Association for Computational Linguistics.

Klim Zaporojets, Lucie-Aimée Kaffee, Johannes Deleu, Thomas Demeester, Chris Develder, and Isabelle Augenstein. 2022. Tempel: Linking dynamically evolving and newly emerging entities. *Advances in Neural Information Processing Systems*, 35:1850–1866.

Shanshan Zhang, Lihong He, Eduard Dragut, and Slobodan Vucetic. 2019. How to invest my time: Lessons from human-in-the-loop entity extraction. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2305–2313.

Shanshan Zhang, Lihong He, Slobodan Vucetic, and Eduard Dragut. 2018. Regular expression guided entity mention mining from noisy web data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1991–2000, Brussels, Belgium. Association for Computational Linguistics.

Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 50–61, Online. Association for Computational Linguistics.

A More statistics

A.1 Re-annotating documents from SciDMT

Table 4 presents the details of the entities annotation workload of the 100 documents from SciDMT. Specifically, the 100 documents from SciDMT-E original contains 21281 entity annotations. After our re-annotation process, we compare against the previous SciDMT-E entity annotation, we find that we keep 15989 correctly annotated entities, and remove 709 wrongly annotated entities, fixed 4583 entities and add 2651 new entities. Finally, for this 100 publications we derive from SciDMT contains 23223 entity annotations. Totally, we revived 7234 entities.

#Initial	#Correct	# Removed	#Fixed	# Added	# Final
21281	15989	709	4583	2651	23223

Table 4: The details of our entity annotations efforts for the first 100 documents.

A.2 Comparison with SciERC

Table 5 and Table 6 show the label statistics of SciERC when only keep the DATASET, METHOD, and TASK entities. We can find that, though SciERC annotated 500 abstract, there are only 1575 entities and 1575 relations related to DATASET, METHOD, and TASK.

Relation type	#
FEATURE-OF	28
CONJUNCTION	292
USED-FOR	876
COMPARE	78
HYPONYM-OF	154
PART-OF	78
EVALUATION-FOR	69
Total	1575

Table 5: The relation types distribution of datasets (material), methods, and tasks in SciERC.

Dataset	Dataset	Method	Task	Total
SciERC	561	1592	997	1575
SciER	3942	15881	4695	24518

Table 6: The entity distribution of datasets (material), methods, and tasks in SciERC and SciER.

A.3 SciER Statistics

Table 7 provide the label distribution of the train, development, ID test and OOD test of our proposed SciER.

Rel./Ent. Type	Train	Dev	ID Test	OOD Test	Total
DATASET	11424	1549	1890	1018	15881
DATASET	3220	269	370	83	3942
TASK	3397	416	688	194	4695
Total	18041	2234	2948	1295	24518
PART-OF	1865	214	304	111	2494
USED-FOR	2398	343	546	167	3454
EVALUATED-WITH	863	78	131	49	1121
SYNONYM-OF	880	76	170	89	1215
COMPARE-WITH	875	175	114	54	1218
SUBCLASS-OF	697	114	176	73	1060
BENCHMARK-FOR	551	64	85	28	728
SUBTASK-OF	210	31	65	9	315
TRAINED-WITH	404	37	35	2	478
Total	8743	1132	1626	582	12083

Table 7: The label distribution of our SciER.

B More Implementation Details

B.1 Supervised Baselines

We followed the hyperparameter settings recommended in the PURE, PL-Maker, and HGERE papers respectively. All experiments were conducted using two NVIDIA A100 80GB GPUs for training. All reported experimental results represent the average of five runs, each with a different random seed.

Hyperparameter	GPT-3.5-Turbo	Llama3-70b	Qwen2-72b
Engine	gpt-3.5-turbo-0125	Llama3-70b-instruct	Qwen2-72b-instruct
Temperature	0.3	0.3	0.3
Max_tokens	256	256	256
Top_p	0.9	0.9	0.9

Table 8: Hyperparamters of GPT-3.5-turbo, Llama3-70b and Qwen2-72b.

B.2 LLM-based Baselines

The hyperparameters of GPT-3.5-turbo, Llama3-70b, and Qwen2-72b are presented in the Table 8. The used version of SimCSE is *sup-simcse-roberta-large*[5]. To ensure fairness in the comparison, we kept the inference hyperparameters consistent for both models. For the GPT-3.5-turbo experiments, due to cost considerations, we sampled 200 sentences from each test set for testing, conducted the tests three times, and then averaged the results. The total cost of GPT-3.5-turbo experiments are 50.25 dollars.

For the Llama3-70b and Qwen2-72b, we used two NVIDIA A100 80GB GPUs for inference. We tested on all samples in each test set, conducted the tests five times, and then averaged the results. Due to the input length limitation of Llama3-70b

[5] https://huggingface.co/princeton-nlp/sup-simcse-roberta-large

and the lengths of our prompt templates, we set the number of demonstrations for each task as 30, which is also recommended by recent GPT-3 based relation extraction work (Wan et al., 2023).

C More analysis

C.1 Qualitative Example

Table 9 shows one OOD test example for different models. We observe that both PL-Marker and HGERE fail on this example due to the NER results. PL-Marker ignores the TASK "therapeutic molecular generation", and HGERE predicts the wrong span. But if we provide the gold standard entities to PL-Marker, i.e., the PL-Marker (RE) . It predict correctly. All LLMs-based baselines perform well on this example.

C.2 Relation Extraction Statistic

We present the proportion of NULL categories in the RE task in the table 10. We found that the proportion exceeds 60%.

D Prompt Design

In this section, we provide the details of annotation guideline-enhanced prompt designs for each task. We list the few-shot version of NER, RE, and Joint ERE. To save the space, we only keep provide 1 demonstration for each task. In our experiments, we use 30 demonstrations. All the zero-shot version are just removed the demonstrations.

Few-Shot NER

```
### Task: Generate an HTML version of an
input text, marking up specific entities
related to machine learning and
artificial intelligence. The entities to
be identified are: 'Dataset', 'Task',
and 'Method'. Use HTML <span> tags to
highlight these entities. Each <span>
should have a class attribute indicating
the type of the entity.

### Entity Definitions:
- 'Task': A task in machine learning
refers to the specific problem or type
of problem that a ML/AI model/method is
designed to solve. Tasks can be broad,
like classification, regression, or
clustering, or they can be very
specific, such as Pedestrian Detection,
Autonomous Driving, Sentiment Analysis,
Named Entity Recognition and Relation
Extraction...
- 'Method': A method entity refers to
the approach, algorithm, or technique
used to solve a specific task/problem.
Methods encompass the computational
algorithms, model architectures, and the
training procedures that are employed to
make predictions or decisions based on
data. For example, Convolutional Neural
Networks, Dropout, data augmentation,
recurrent neural networks...
- 'Dataset': A realistic collection of
data that is used for training,
validating, or testing the algorithms.
These datasets can consist of various
forms of data such as text, images,
videos, or structured data. For example,
MNIST, COCO, AGNews, IMDb...

### Entity Markup Guide:
- Use <span class="Task"> to denote a
Task entity.
- Use <span class="Method"> to denote a
Method entity.
- Use <span class="Dataset"> to denote a
Dataset entity.

### Other Notes:
- Generics cannot be used independently
to refer to any specific entities, e.g.,
'This task', 'the dataset', and 'a
public corpus' are not entities.
- The determiners should not be part of
an entity span. For example, given span
'the SQuAD v1.1 dataset', where the
determiner 'the' should be excluded the
entity span.
- If both the full name and the
abbreviation are present in the
sentence, annotate the abbreviation and
its corresponding full name separately.
For instance, '20-newsgroup ( 20NG )',
the annoation should be '<span
class="Dataset">20-newsgroup</span> (
<span class="Dataset">20NG</span> )'.
- If one entity with exact same span text
appears many times within a sentence,
all span text should be marked up.
- If one sentence without any entities
appear, do not mark up any span text.
- Only annotate "factual,
content-bearing" entities. Task,
dataset, and method entities normally
have specific names and their meanings
```

	Example
Ground Truth	Figure 5 shows the process undertaken by **GxVAEs** for **therapeutic molecular generation** . (USED-FOR)
PL-Marker	Figure 5 shows the process undertaken by **GxVAEs** for therapeutic molecular generation.
HGERE	Figure 5 shows the process undertaken by **GxVAEs** for therapeutic **molecular generation** . (USED-FOR)
PL-Marker (RE)	Figure 5 shows the process undertaken by **GxVAEs** for **therapeutic molecular generation** . (USED-FOR)
Llama3-70b (joint)	Figure 5 shows the process undertaken by **GxVAEs** for **therapeutic molecular generation** . (USED-FOR)
GPT-3.5-Turbo(Joint)	Figure 5 shows the process undertaken by **GxVAEs** for **therapeutic molecular generation** . (USED-FOR)
Llama3-70b (pipeline)	Figure 5 shows the process undertaken by **GxVAEs** for **therapeutic molecular generation** . (USED-FOR)
GPT-3.5-Turbo(pipeline)	Figure 5 shows the process undertaken by **GxVAEs** for **therapeutic molecular generation** . (USED-FOR)

Table 9: Test results of one OOD test example with PL-Marker, HGERE, Llama3-70b (joint), GPT-3.5-Turbo (joint), Llama3-70b (pipeline), GPT-3.5-Turbo (pipeline). The PL-Marker (RE) means using PL-Marker to predict the relation with given two entities.

	# relation	# NULL	Tot,	NULL (%)
ID test set	1626	4715	6341	74.46%
OOD test set	582	1109	1691	65.58%
Dev	1132	2053	3185	64.46%
Train	8743	20923	29666	70.53%

Table 10: Statistics of datasets for relation extraction. "NULL" means the given subject and object pairs do not have relation.

are consistent across different papers. For example, the "CoNLL03", "SNLI" are factual entities.
- Minimum span principle. Annotators should annotate only the minimum span necessary to represent the original meaning of task/dataset/metric (e.g.: "The", "dataset", "public", 'method', 'technique' are often omitted).

Examples:
Input: In particular we briefly introduce the principal concepts behind deep Convolutional Neural Networks (CNNs) , describe the architectures used in our analysis and the algorithms adopted to train and apply them .
Output: In particular we briefly introduce the principal concepts behind deep Convolutional Neural Networks (CNNs) , describe the architectures used in our analysis and the algorithms adopted to train and apply them .

Input: Specifically , we investigate the attention and feature extraction mechanisms of state - of - the - art recurrent neural networks and self - attentive architectures for sentiment analysis , entailment and machine translation under adversarial attacks .

Output:

Few-Shot RE

Task: Based on the given sentence, and subject entity and object entity from the sentence, answer the questions to determine the relationship between them. The potential relations are: ['Part-Of', 'SubClass-Of', 'SubTask-Of', 'Benchmark-For', 'Trained-With', 'Evaluated-With', 'Synonym-Of', 'Used-For' , 'Compare-With']. Answer 'NULL' to indicate that there is no relationship between the entities.

Relationship Definitions:
- 'Part-Of': This relationship denotes that one method is a component or a part of another method.
- 'SubClass-Of': Specifies that one method is a subclass or a specialized version of another method.
- 'SubTask-Of': Indicates that one task

is a subset or a specific aspect of another broader task.
- 'Benchmark-For': Shows that a dataset serves as a standard or benchmark for evaluating the performance of methods on a specific task.
- 'Trained-With': Indicates that a method is trained using a specific dataset.
- 'Evaluated-With': This relationship denotes that a method is evaluated using a specific dataset to test its performance or conduct the experiments.
- 'Synonym-Of': Indicates that two terms or entities are considered to have the same or very similar meaning, such as abbreviation.
- 'Used-For': Shows that one entity is utilized for achieving or performing another entity. For example, one Method is Used-For one Task. This relationship is highly flexible, allowing for generic relationships across diverse entities.
- 'Compare-With': This relationship is used when one entity is compared with another to highlight differences, similarities, or both.

Notes:
- Determine the 'Relationship' that best describes how the entities are related, or just answer 'NULL' if no relationship exists.
- Please do not annotate negative relations. For example, X is not used in Y or X is hard to be applied in Y.
- Annotate a relationship only if there is direct evidence or clear implication in the text. Avoid inferring relationships that are not explicitly mentioned or clearly implied.

Examples:
Input: In particular we briefly introduce the principal concepts behind deep Convolutional Neural Networks (CNNs) , describe the architectures used in our analysis and the algorithms adopted to train and apply them .
Subject Entity: Convolutional Neural Network
Object Entity: CNNs
Output: Synonym-Of

Input: Specifically , we investigate the attention and feature extraction mechanisms of state - of - the - art recurrent neural networks and self - attentive architectures for sentiment analysis , entailment and machine translation under adversarial attacks .
Subject Entity: attention
Object Entity: feature extraction mechanisms

Output:

Few-Shot Joint ERE

Task: Identify and extract all relationship triplets consisting of two entities and their relationship from the input text. Each triplet consists of one subject entity, one object entity and their relationship. The interested entity types are: ['Dataset', 'Method', 'Task']. The potential relations are: ['Part-Of', 'SubClass-Of', 'SubTask-Of', 'Benchmark-For', 'Trained-With', 'Evaluated-With', 'Synonym-Of', 'Used-For' , 'Compare-With']. Answer 'NULL' to indicate that there is no triplet.

Entity Definitions:
- 'Task': A task in machine learning refers to the specific problem or type of problem that a ML/AI model/method is designed to solve. Tasks can be broad, like classification, regression, or clustering, or they can be very specific, such as Pedestrian Detection, Autonomous Driving, Sentiment Analysis, Named Entity Recognition and Relation Extraction...
- 'Method': A method entity refers to the approach, algorithm, or technique used to solve a specific task/problem. Methods encompass the computational algorithms, model architectures, and the training procedures that are employed to make predictions or decisions based on data. For example, Convolutional Neural Networks, Dropout, data augmentation, recurrent neural networks...
- 'Dataset': A realistic collection of data that is used for training, validating, or testing the algorithms. These datasets can consist of various

forms of data such as text, images, videos, or structured data. For example, MNIST, COCO, AGNews, IMDb...

Relationship Definitions:
- 'Part-Of': This relationship denotes that one method is a component or a part of another method.
- 'SubClass-Of': Specifies that one method is a subclass or a specialized version of another method.
- 'SubTask-Of': Indicates that one task is a subset or a specific aspect of another broader task.
- 'Benchmark-For': Shows that a dataset serves as a standard or benchmark for evaluating the performance of methods on a specific task.
- 'Trained-With': Indicates that a method is trained using a specific dataset.
- 'Evaluated-With': This relationship denotes that a method is evaluated using a specific dataset to test its performance or conduct the experiments.
- 'Synonym-Of': Indicates that two terms or entities are considered to have the same or very similar meaning, such as abbreviation.
- 'Used-For': Shows that one entity is utilized for achieving or performing another entity. For example, one Method is Used-For one Task. This relationship is highly flexible, allowing for generic relationships across diverse entities.
- 'Compare-With': This relationship is used when one entity is compared with another to highlight differences, similarities, or both.

Notes:
- Input sentence has one triplet: [['entity1 span text:entity1 type', 'relationship', 'entity2 span text:entity2 type']]
- Input sentence has no triplets: []
- Annotate a relationship only if there is direct evidence or clear implication in the text. Avoid inferring relationships that are not explicitly mentioned or clearly implied.
- Ensure that the entity spans are exact extracts from the input text and that the relationships accurately reflect the described interactions. Ensure the output is in the correct format (A list of triplets).
- Entities in the triplet should have same form as input sentence.

Examples:
Input: In particular we briefly introduce the principal concepts behind deep Convolutional Neural Networks (CNNs) , describe the architectures used in our analysis and the algorithms adopted to train and apply them .
Output: [['CNNs:Method', 'Synonym-Of', 'Convolutional Neural Networks:Method']]

Input: Specifically , we investigate the attention and feature extraction mechanisms of state - of - the - art recurrent neural networks and self - attentive architectures for sentiment analysis , entailment and machine translation under adversarial attacks .
Subject Entity: attention
Object Entity: feature extraction mechanisms

Output:

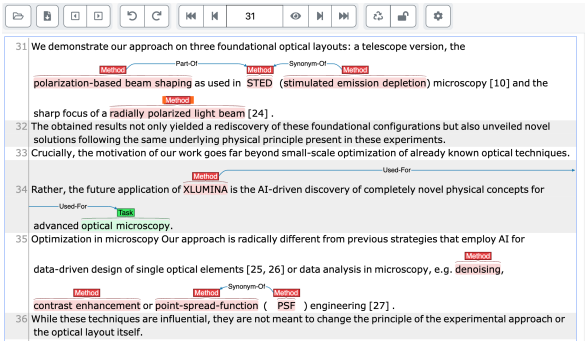


Figure 5: Annotation interface.

E Annotation Guideline

This section contains the basic information from our annotation guideline for double-blind review.

E.1 Annotation Tool

We use the INCEpTION[6] as our annotation platform. Figure 5 shows our annotation interface.

E.2 Entity Annotation

Scientific entities in the machine learning (ML) or Artificial intelligence (AI) domains refer to key

[6] https://github.com/inception-project/inception

concepts or components that are integral to the structure and study of ML/AI papers. We follow the definition of entities/terms and build our annotation guides for NER based on the ACL RD-TEC Annotation Guideline (QasemiZadeh and Schumann, 2016), Papers With Code (PwC) and SciDMT (Pan et al., 2024b). We are interested in three specific entity types: Dataset, Task, and Method.

Dataset: A realistic collection of data that is used for training, validating, or testing the algorithms. These datasets can consist of various forms of data such as text, images, videos, or structured data. For example, MNIST, COCO, AGNews, IMDb, etc.

Task: A task in machine learning refers to the specific problem or type of problem that a ML/AI model is designed to solve. Tasks can be broad, like classification, regression, or clustering, or they can be very specific, such as Pedestrian Detection, Autonomous Driving, Sentiment Analysis, Named Entity Recognition and Relation Extraction.

Method: A method entity refers to the approach, algorithm, or technique used to solve a specific task/problem. Methods encompass the computational algorithms, model architectures, and the training procedures that are employed to make predictions or decisions based on data. For example, Convolutional Neural Networks (CNNs),

Annotation Notes:

Considering that annotators may have varying understandings of the annotation details, we have defined a set of rules and notes to standardize the annotation process:

- Do not annotate generics and determiners. Generics cannot be used independently to refer to any specific entities, e.g., “This task”, “the dataset”, “a public corpus” etc. The determiners should not be part of an entity span. For example, given span "the SQuAD v1.1 dataset", where the determiner “the” should be excluded the entity span. We refer ignoring.

- Minimum span principle. Annotators should annotate only the minimum span necessary to represent the original meaning of task/dataset/metric (e.g.: "The", "dataset", "public", ‘method’, ‘technique’ are often omitted).

- Only annotate “factual, content-bearing” entities. Task, dataset, and method entities normally have specific names and their meanings are consistent across different papers. For example, the “CoNLL03”, “SNLI” are factual entities.

- If one entity with exact same span text appears many times within a sentence, all span text should be annotated.

E.3 Relation Annotation

Relation links cannot exceed the sentence boundary. We define 9 types of relations for Dataset, Method, and Task entities.

Relation Definitions:

- ’Part-Of’: This relationship denotes that one method is a component or a part of another method.

- ’SubClass-Of’: Specifies that one method is a subclass or a specialized version of another method.

- ’SubTask-Of’: Indicates that one task is a subset or a specific aspect of another broader task.

- ’Benchmark-For’: Shows that a dataset serves as a standard or benchmark for evaluating the performance of methods on a specific task.

- ’Trained-With’: Indicates that a method is trained using a specific dataset.

- ’Evaluated-With’: This relationship denotes that a method is evaluated using a specific dataset to test its performance or conduct the experiments.

- ’Synonym-Of’: Indicates that two terms or entities are considered to have the same or very similar meaning, such as abbreviation.

- ’Used-For’: Shows that one entity is utilized for achieving or performing another entity. For example, one Method is Used-For one Task. This relationship is highly flexible, allowing for generic relationships across diverse entities.

- ’Compare-With’: This relationship is used when one entity is compared with another to highlight differences, similarities, or both.

Annotation Notes:

- Do not annotate negative relations. For example, X is not used in Y or X is hard to be applied in Y.

- Verify that the entities involved in the relation match the prescribed types (e.g., Method-Dataset for Trained-With). Incorrect entity types should not be linked by these specific relations.

- Annotate a relationship only if there is direct evidence or clear implication in the text. Avoid inferring relationships that are not explicitly mentioned or clearly implied.

- Ensure consistency in how relationships are annotated across different texts. If uncertain, refer back to the guideline definitions or consult with a supervisor.

- Do not make assumptions about relationships based on personal knowledge or external information. Rely solely on the information provided in the text.