Extracting Machine Learning Knowledge from Scholarly Publications:

From Fine-Grained Information Extraction to Cross-Schema Generalization

Inaugural-Dissertation

zur

Erlangung des Doktorgrades der
Mathematisch-Naturwissenschaftlichen Fakultät
der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

Wolfgang Otto

aus Hildesheim

Düsseldorf, Juni 2026

aus dem Institut für Informatik der
Heinrich-Heine-Universität Düsseldorf

Gedruckt mit der Genehmigung der
Mathematisch-Naturwissenschaftlichen Fakultät der
Heinrich-Heine-Universität Düsseldorf

Berichterstatter:

1. Prof. Dr. Stefan Dietze
2. Prof. Dr. Stefan Conrad

Tag der mündlichen Prüfung: 2. September 2026

Erklärung

Ich versichere an Eides Statt, dass die Dissertation von mir selbstständig und ohne unzulässige fremde Hilfe unter Beachtung der Grundsätze zur Sicherung guter wissenschaftlicher Praxis an der Heinrich-Heine-Universität Düsseldorf erstellt worden ist. Die hier vorgelegte Dissertation habe ich eigenständig und ohne unerlaubte Hilfe angefertigt. Die Dissertation wurde in der vorgelegten oder in ähnlicher Form noch bei keiner anderen Institution eingereicht. Ich habe bisher keine erfolglosen oder erfolgreichen Promotionsversuche unternommen.

Düsseldorf, Deutschland — Wolfgang Otto
16. Juni 2026

Hinweis zur Nutzung KI-basierter Hilfsmittel Eine Dokumentation der im Rahmen der Erstellung dieser Dissertation genutzten KI-basierten Hilfsmittel findet sich in Anhang B.

Acknowledgements

This dissertation was written within a working environment that has shaped both my research and my personal development. At GESIS, and especially within KTS, I found a place that combines professional standards and close scientific exchange with a friendly and supportive atmosphere. I am grateful for the opportunities, trust, and freedom I received there.

My deepest academic gratitude goes to my doctoral supervisor, Prof. Dr. Stefan Dietze, who believed in me and my research from the beginning. He gave me the freedom to develop my own ideas while guiding me towards a coherent research profile and a consistent body of work. His confidence in my abilities was essential throughout this process. I am also grateful to Prof. Dr. Stefan Conrad for his reliability, responsiveness, and support.

I thank the team leads, Philipp, Dimi, and Peter, as well as all my colleagues, for giving me the room to take responsibility, explore new directions, and grow into my role as a researcher. I especially thank Dimi for regularly helping me place my ambitions into a realistic perspective and for reminding me, when necessary, to focus on what could and should be achieved.

A special acknowledgement goes to the GSAP team, our focus group on scholarly AI, which accompanied and supported my work over several years. The team provided the setting in which many of the ideas behind this dissertation were discussed, challenged, revised, and finally put into practice. I especially thank Matthäus and Lu for the intensive and, at times, exhausting discussions, and for their persistence in continuing them until we reached a shared understanding. I am particularly grateful to Matthäus for allowing me to learn from his expertise and for becoming not only a valued colleague but also a close friend. I also thank all the students with whom I had the pleasure of working. Without the shared commitment and team spirit of this group, these words would never have been written.

I also thank the many other colleagues and scientific friends who supported me through detailed discussions and open exchange. These conversations ranged from implementation decisions up to philosophical mindgames about artificial intelligence, research, and scientific practice. Stephan and Sebastian are named here as representatives of the many people whose ideas, criticism, humour, and curiosity made these discussions valuable.

Looking further back in my academic career, I thank all the supervisors who encouraged me to believe in independent research based on mutual trust. I am especially grateful to Prof. Gerhard Heyer, who motivated me to focus on natural language processing, gave me the opportunity to work in my first research project, and supervised

my bachelor's and master's theses in this field. He introduced me to an area in which I have continued to learn and work with lasting interest. I also thank him for trusting me to pursue an academic career at a time when this path was far from certain.

Beyond my academic and professional environment, I thank my long-term friends. I am grateful to all those who have remained part of my life over many years and across different places and phases. Knowing that I can rely on each of them has given me stability far beyond the work described in this dissertation.

I thank my parents for believing in me and for giving me the confidence and freedom to make my own decisions. Their support enabled me to follow not only direct routes but also necessary detours, without having to fear that an uncertain decision would leave me without support.

My most personal thanks go to my loved ones. I thank Marie for trying to handle me while I tried to handle research, deadlines, and unfinished texts. We both succeeded. I thank Eddy for his patience while waiting for me to finally leave the screen and turn to what was most important at that moment: our game.

In closing, I would like to thank Micha for showing me that independence and self-assurance do not require following the path expected by others, and that the freedom to choose one's own way can lead to remarkable outcomes and shape a remarkable character.

Zusammenfassung

Die wachsende Zahl wissenschaftlicher Publikationen und der zunehmende Einsatz von maschinellem Lernen (ML) in vielen Fachdisziplinen erschweren es, technische Informationen aus der Literatur manuell zu erfassen, zu vergleichen und wiederzuverwenden. In Publikationen zum maschinellen Lernen sind Informationen zu Modellen, Datensätzen, Tasks, Software, Baselines und experimentellen Vergleichen häufig über den gesamten Volltext verteilt.

Diese Dissertation geht davon aus, dass wissenschaftliche Informationsextraktion für Literatur zum maschinellen Lernen mehr leisten muss, als Textstellen zu markieren. Sie muss erfassen, welche Rolle ML-Artefakte in Experimenten spielen, auf welche Textstellen sich diese Informationen stützen und wie unterschiedliche Annotations- und Extraktionsschemata zueinander stehen. Die Arbeit untersucht daher, wie ML-Artefakte und ihre Beziehungen als strukturierte, beleggestützte Repräsentationen aus Volltextpublikationen extrahiert werden können. Außerdem untersucht sie, wie unterschiedliche Ressourcen und Schemata miteinander vergleichbar gemacht werden können, ohne ihre jeweiligen Perspektiven auf wissenschaftliche Information vollständig zu vereinheitlichen.

Das Ziel der Dissertation ist die Entwicklung und Bewertung von Methoden zur wissenschaftlichen Informationsextraktion (IE) für ML-Artefakte und ihre Beziehungen in Volltextpublikationen. Im Mittelpunkt steht nicht nur, ob ein Artefakt im Text erwähnt wird, sondern auch, welche Art von Artefakt genau gemeint ist und wie es mit anderen Artefakten im Experiment zusammenhängt. Dazu unterscheidet die Arbeit unter anderem zwischen Modellarchitekturen und konkreten Modellinstanzen, zwischen Datensätzen und Rohdatenquellen sowie zwischen benannten Artefakten und generischen Artefakterwähnungen. Diese Unterscheidungen machen sichtbar, welche Funktion ein Artefakt im jeweiligen experimentellen Zusammenhang hat.

Die Dissertation leistet fünf Beiträge, die aufeinander aufbauen und von feingranularen Extraktionsressourcen über schemaübergreifende Generalisierung bis zur interaktiven Nutzung führen. Erstens führt sie GSAP-NER ein, ein feingranulares Entitätsschema und ein manuell annotiertes Volltextkorpus für ML-bezogene Entitäten in Informatikpublikationen, zusammen mit Baseline-Modellen und Analysen zu Annotation und Modellierung. Zweitens untersucht sie die LLM-gestützte Extraktion von Softwareinformationen. Dazu wird der Extraktions-Task als Single-Choice Question Answering reformuliert. Außerdem wird retrieval-basiertes Prompting bei begrenzter Supervision evaluiert. Drittens erweitert sie die Entitätsebene von GSAP-NER zu GSAP-ERE, einer Volltextressource für Entity and Relation Extraction (ERE). Diese Ressource modelliert Workflow-Relationen zwischen ML-Artefakten, darunter Verknüpfungen zwischen

Modellen, Methoden, Datensätzen, Tasks, Eigenschaften, Vergleichen und Referenzen. Viertens untersucht sie schemaübergreifende Generalisierung über wissenschaftliche ERE-Benchmarks hinweg. Dazu werden verwandte Datensätze zunächst in einem gemeinsamen Evaluationsrahmen vergleichbar gemacht. Die Evaluation berücksichtigt sowohl einen vereinheitlichten Labelraum als auch die ursprünglichen datensatzspezifischen Label- und Vorhersageräume. Anschließend werden verschiedene Modell- und Datensatzkombinationen in einem schemaerhaltenden Multi-Dataset-Setup trainiert und evaluiert, ohne die individuellen Schemata der Datensätze aufzugeben. Fünftens stellt sie Methods Miner vor, ein nutzerorientiertes System, das die Extraktionspipeline auf Artikelsammlungen anwendet und Nutzern ermöglicht, extrahierte ML-Artefakte zusammen mit ihren Belegen auf Satzebene zu prüfen.

Die Ergebnisse zeigen, dass ML-Artefakte mit einem feingranularen Typsystem aus Volltextpublikationen extrahiert werden können. Diese Extraktion bleibt jedoch durch Mehrdeutigkeit bei konzeptionellen Entitätstypen, Modellverweisen, generischen Erwähnungen und Relationsextraktion begrenzt. Die Ergebnisse zeigen außerdem, wie Relationen ermöglichen, von Listen erwähnter Artefakte zu Darstellungen experimenteller Abläufe zu gelangen. Anhand wissenschaftlicher ERE-Benchmarks zeigt die Dissertation, dass Schema-Abgleich kontrollierte Vergleiche ermöglicht, aber eine einfache Schema-Zusammenführung nicht für robusten Transfer ausreicht. Ein schemaerhaltender Multi-Datensatz-Ansatz ist daher besser geeignet, wenn Datensätze zwar ähnliche wissenschaftliche Ziele verfolgen, sich aber in Annotationsdichte und -stil, Korpusauswahl, Label-Granularität und Relationssemantik unterscheiden. Damit stellt die Dissertation Datensätze, Schemata, Modelle, Evaluationsaufbauten und ein Prototyp-System bereit, die eine nachvollziehbarere Extraktion, Bewertung und Exploration von Wissen zum maschinellen Lernen in wissenschaftlichen Publikationen unterstützen.

Abstract

The growing number of scholarly publications and the increasing use of machine learning and artificial intelligence across scientific disciplines make it difficult to manually identify, compare, and reuse technical information from the literature. In machine-learning publications, information about models, datasets, tasks, software, baselines, and experimental comparisons is often distributed across the full texts rather than stated in one place.

This thesis is guided by the view that scholarly information extraction (IE) for machine-learning (ML) literature must preserve fine-grained artifact distinctions, links to textual evidence, and awareness of schema differences across annotation and extraction projects. It investigates how such knowledge can be extracted as evidence-linked records of ML artifacts and relations. It further studies how such records can be compared across annotation schemes and extraction settings, and how extraction models can transfer across datasets without collapsing dataset-specific views.

The central goal of the thesis is to develop and evaluate scholarly information extraction methods for machine-learning artifacts and their typed relations in full-text publications. Extraction is treated not only as the detection of text spans, but as the construction of structured records that remain traceable to the sentences from which they were derived. These records distinguish artifact roles that are often merged in broader scientific IE schemas, such as model architectures versus concrete model instances, datasets versus raw-data sources, and named artifacts versus generic artifact mentions. Such distinctions are important because these objects can play different roles in experimental workflows.

The thesis makes five contributions that build on each other, moving from fine-grained extraction resources to cross-schema generalization and interactive use. First, it introduces GSAP-NER, a fine-grained entity schema and manually annotated full-text corpus for ML-related entities in computer science publications, together with baseline models and analyses of annotation and modeling challenges. Second, it studies extraction methods based on large language models by reformulating software-related extraction as single-choice question answering and evaluating retrieval-based prompting in a limited-supervision setting. Third, it extends the entity layer with GSAP-ERE, a full-text entity and relation extraction (ERE) resource that models workflow relations between ML artifacts, including links between models, methods, datasets, tasks, properties, comparisons, and references. Fourth, it investigates cross-schema generalization across scientific ERE benchmarks. Related datasets are first made comparable in a shared evaluation framework that considers both a unified label space and the original dataset-specific label and prediction spaces. Different model and dataset combinations are then

trained and evaluated in a schema-preserving multi-dataset setup without giving up the individual schemas of the datasets. Fifth, it presents Methods Miner, a user-facing system that applies the extraction pipeline to paper collections and lets users inspect extracted ML artifacts together with their sentence-level evidence.

The results show that fine-grained extraction of ML artifacts from full-text publications is feasible, but remains limited by ambiguity in conceptual entity types, model references, generic mentions, and relation extraction. They also show that relations are necessary to move from lists of mentioned artifacts to workflow-level representations of experiments. Across scientific ERE benchmarks, the thesis shows that schema alignment enables controlled comparison, but that simple schema unification is not sufficient for robust transfer. A schema-preserving multi-dataset setup is therefore more suitable when datasets share broad scientific targets but differ in annotation density, corpus selection, label granularity, and relation semantics.

Taken together, the results show that artifact granularity, evidence linking, and schema-aware evaluation interact. Fine-grained artifact distinctions improve the usefulness of extracted records only when they remain linked to textual evidence, and schema-aware evaluation is needed to compare such records or transfer extraction models across datasets without removing relevant distinctions. This combination makes extracted records more useful for evaluation, comparison, and interactive exploration, because users can move from aggregated views back to the sentences that support them. At the same time, the thesis identifies important limits of text-only, sentence-level extraction, including cross-sentence relations, document-level coreference, artifact normalization, table and figure evidence, and corpus-wide entity linking. The thesis therefore provides datasets, schemas, models, evaluation setups, and a prototype system as a basis for more reliable extraction and analysis of machine-learning knowledge in scholarly publications.

Contents

Erklärung **i**

Acknowledgements **iii**

Zusammenfassung **v**

Abstract **vii**

List of Figures **xiii**

List of Tables **xvi**

List of Prompts **xvii**

List of Abbreviations **xix**

1 Introduction **1**
1.1 Context and Motivation 2
1.2 Problem Statement and Research Goal 3
1.3 Contributions of this Thesis 6
1.4 Contributing Publications 8
1.5 Outline 17

2 Background: IE, Scientific NLP, and Scholarly IE **19**
2.1 Information Extraction Tasks and Settings 19
2.2 Scholarly Information Extraction and Scientific NLP 22
2.3 Extraction Targets in Scholarly Publications 23
2.4 Resources and Benchmarks for Scholarly IE 25
2.5 Modeling Approaches and LLM-Based Extraction 27
2.6 Annotation, Evaluation, and Schema Mismatch 29
2.7 Applications and Exploration Systems 31

3 Fine-Grained Entity Recognition for ML Research **35**
3.1 Introduction 36
3.2 Related Work 37
3.3 Dataset 39
3.4 Baseline Model 44

3.5 Experimental Setup . . . 45
3.6 Experimental Results . . . 47
3.7 Conclusion . . . 48

4 LLM-Assisted Extraction of Software Information 51
4.1 Introduction . . . 52
4.2 Related Work . . . 52
4.3 SOMD Shared Task . . . 53
4.4 Using GLMs for Software-Related IE Tasks . . . 54
4.5 Experiments . . . 56
4.6 Results . . . 58
4.7 Conclusion . . . 59
4.8 Chapter Appendix . . . 59

5 Modeling Relations between ML Artifacts 67
5.1 Introduction . . . 68
5.2 Related Work . . . 70
5.3 Preliminaries . . . 72
5.4 GSAP-ERE Dataset . . . 72
5.5 Experiments . . . 75
5.6 Experimental Results . . . 79
5.7 Conclusion . . . 80
5.8 Limitations . . . 80

6 Schema-Preserving Generalization Across Scientific ERE Benchmarks 83
6.1 Introduction . . . 84
6.2 Related Work . . . 86
6.3 Dataset Analysis . . . 87
6.4 Experimental Setup . . . 89
6.5 Results . . . 91
6.6 Discussion . . . 94
6.7 Conclusion . . . 95
6.8 Chapter Appendix . . . 97

7 Exploration and Analysis of Extracted ML Knowledge 113
7.1 Introduction . . . 114
7.2 System Description . . . 115
7.3 Demonstration Scenario . . . 117
7.4 Related Work . . . 118
7.5 Conclusion and Future Work . . . 119

8 Conclusion and Future Work 121
8.1 Summary of Contributions . . . 121
8.2 Answers to the Research Questions . . . 122
8.3 Limitations . . . 125
8.4 Future Work . . . 127
8.5 Closing Remarks . . . 128

Appendix **129**
A Full List of Contributions 129
B Documentation of AI-Based Tool Use 135

Bibliography **137**

List of Figures

1.1 Overview of the thesis. 7

3.1 Example annotations of ML model and dataset mentions. 36
3.2 Exact-match F1 of SciBERT by number of training publications. 48

6.1 Corpus-level Dataset Explorer view. 84
6.2 Dataset-conditioned multi-head architecture for schema-preserving ERE. 85
6.3 Temporal SciNLP annotation-density view in the companion application. 102
6.4 GSAP-ERE schema examples. 107
6.5 SciNLP and SciER schema examples. 108

7.1 Screenshot of the Methods Miner. 114
7.2 Top-down system overview of Methods Miner. 116

List of Tables

1.1 Thesis line across chapters, research questions, and contributions 17

3.1 Comparison of scholarly NER datasets, annotation units, and entity types 38
3.2 Annotation examples for the fine-grained ML related entity types. . . . 39
3.3 Interrater agreement for GSAP-NER entity annotations. 42
3.4 Entity span statistics for the GSAP-NER corpus. 43
3.5 F1 scores of fine-tuned PLM baselines for GSAP-NER. 45

4.1 Model and retrieval results for Subtask 1 57
4.2 SOMD performance rankings 58
4.3 Object Properties, Domain, and Range Classes with Inner Software Relations 61
4.4 RAG retrieval example using entity similarity 62
4.5 RAG retrieval example using sentence similarity 62

5.1 Comparison of GSAP-ERE with scientific NER and RE datasets. . . . 69
5.2 GSAP-ERE semantic relation groups. 71
5.3 Domain and range of the 18 relation types. 74
5.4 Baseline F1 scores for GSAP-ERE. 75
5.5 Test-set micro-F1 comparison of baseline approaches. 76
5.6 NER performance by shot number and example selection. 78
5.7 Qwen2.5-32B RE validation micro-F1 scores. 78

6.1 Comparison of scientific ERE datasets 87
6.2 Final hyperparameters used for training. 89
6.3 Reproduced dataset-baseline micro-F1 scores. 90
6.4 Cross-dataset and unified-training micro-F1 scores. 92
6.5 MultiSciERE entity and relation micro-F1 scores by test set. 93
6.6 Entity and relation retention after applying the full unification pipeline. Counts are aggregated across train, development, and test splits. 98
6.7 Entity retention across the unification pipeline. 99
6.8 Relation retention across the unification pipeline. 100
6.9 Entity labels removed because they have no stable counterpart in the unified schema. 100
6.10 Entity confusion matrix for SciER transfer to GSAP-ERE. 101
6.11 Entity label alignment used for unified evaluation. 104

6.12 Relation label alignment used for unified evaluation. Example sentences are illustrative and are not taken from the datasets. 106
6.13 Entity cross-dataset evaluation results on the test splits of the three datasets. Scores are F1 in percent. Italic values mark same-dataset evaluation. Bold values mark the best cross-dataset score in each row. . 109
6.14 Relation cross-dataset evaluation results on the test splits of the three datasets. Scores are F1 in percent. Italic values mark same-dataset evaluation. Bold values mark the best cross-dataset score in each row. . 110
6.15 Document counts by cluster and original split. 111
6.16 Paper-wise SciER split for non-NLP to NLP domain-shift evaluation. . 111
6.17 Experimental conditions for the SciER NLP domain-shift experiment. 112
6.18 NER results for the SciER non-NLP to NLP domain-shift experiment. 112
6.19 RE results for the SciER non-NLP to NLP domain-shift experiment. . 112

List of Prompts

Chapter 1.
LaTeX Warning: Float too large for page by 0.44905pt on input line 269.
Chapter 2. Chapter 3. Chapter 4. Chapter 5. Chapter 6.
LaTeX Font Warning: Font shape 'T1/lmr/bx/sc' undefined (Font) using 'T1/lmr/bx/n' instead on input line 179.
Chapter 7. Chapter 8.
LaTeX Font Warning: Some font shapes were not available, defaults substituted.

List of Abbreviations

AI	artificial intelligence.
ERE	entity and relation extraction.
F1	harmonic mean of precision and recall.
GROBID	GeneRation Of BIbliographic Data.
GSAP	Gesis Scholarly Annotation Project.
GSAP-ERE	GSAP entity and relation extraction.
GSAP-NER	GSAP named entity recognition.
IAA	inter-annotator agreement.
IE	information extraction.
KG	knowledge graph.
LLM	large language model.
ML	machine learning.
NER	named entity recognition.
NLP	natural language processing.
OOD	out-of-distribution.
PLM	pretrained language model.
QA	question answering.
RAG	retrieval-augmented generation.
RE	relation extraction.
SciER	scientific entity and relation extraction benchmark.
SciERC	scientific entity, relation, and coreference dataset.

SciNLP	scientific NLP benchmark for entity and relation extraction.
SciREX	scientific information extraction benchmark.
SOMD	software mention detection shared task.
TDMS	task, dataset, metric, and score.

1

Introduction

Imagine reading a machine-learning paper. Understanding such a paper typically means reconstructing its experiment: which data are selected and processed, which models are designed, reused, trained, or evaluated, which task is addressed, and which comparisons support the reported claims. This reconstruction is not a sentence-by-sentence reading of the paper, but an interpretation of how its artifacts, tasks, methods, and results relate to each other. It depends on your background knowledge, your familiarity with related work, and the reason why you read the paper in the first place.

Two readers may therefore build different views of the same paper. A researcher trying to reproduce an experiment may focus on data splits, preprocessing, software, and hyperparameters. A researcher writing a survey may focus on tasks, model families, datasets, and reported comparisons. These views are constrained by the same text and by shared scientific terminology, yet they are not identical, as each is shaped by their differing goals.

Annotation projects make such reconstruction explicit and operational. Annotators and lead annotators negotiate a shared understanding of what should be annotated, which distinctions matter, and how ambiguous cases should be handled. The resulting guideline, label inventory, and adjudicated examples define one operational view of how the unannotated paper text should be interpreted for the annotation task. Automatic systems trained on such data then learn to reproduce this view, including its useful distinctions and its limits. For this reason, scholarly information extraction (IE) is not only a problem of finding spans in text, but also a problem of producing structured, evidence-linked records that can be compared, reused, and related across settings.

This thesis studies how machine-learning concepts, artifact mentions, and the semantic relations between them can be extracted from full-text scholarly publications in a way that is fine-grained, traceable back to the supporting text, and comparable across extraction settings. Rather than relying only on broad scholarly IE categories, this thesis distinguishes between model architectures and concrete model instances, datasets and raw-data sources, and named artifacts and generic artifact mentions. Each extracted entity and relation remains linked to the sentence from which it was derived. The thesis first addresses fine-grained entity and relation extraction (ERE), then examines how extraction methods behave across different representational views in scientific IE benchmarks, and finally studies how the resulting records can support comparison, analysis, and exploration of machine-learning literature.

1.1 Context and Motivation

The need for automatic approaches arises from two connected developments: the continued growth of the scholarly record and the increasing use of machine learning across scientific fields. Together, these developments increase both the amount of literature to process and the amount of technical information that researchers must identify, compare, and reuse. This is especially relevant for machine-learning publications, where scientific claims are tied to experimental artifacts such as datasets, models, software, tasks, evaluation settings, baselines, and benchmarks.

Literature growth and ML spread The volume of scientific publications has grown strongly over time, and the effort required for search, triage, and careful reading has grown with it, even within focused research areas (Bornmann and Mutz, 2015; Fortunato et al., 2018). At the same time, machine learning has become a central method family across disciplines (Jordan and Mitchell, 2015), with recent bibliometric studies showing a rapid spread of AI-related research across many scientific fields (Hajkowicz et al., 2023; Bianchini et al., 2025). The success of deep learning methods and the availability of large-scale training data and compute have further contributed to this uptake (LeCun et al., 2015). Together, publication growth and widespread ML use increase both the amount of literature to process and the technical detail that researchers must interpret, including modeling choices, training setups, evaluation claims, and shared artifacts such as pretrained models, benchmark suites, and libraries (Fortunato et al., 2018; Jordan and Mitchell, 2015).

ML publications as artifact-centered texts Machine-learning publications report experimental workflows in which data are selected and processed, models are designed or reused, systems are trained, and results are evaluated against tasks, benchmarks, and baselines. The artifacts involved in these steps determine what reported results mean and whether they can be compared, reused, or reproduced. Incomplete or inconsistent reporting of such details creates friction for reuse and reproducibility-oriented analysis, as documented by reproducibility studies and recent reporting-standard work (Pineau et al., 2021; Raff, 2019; Kapoor et al., 2024). It also affects comparison across papers, because weak baselines, inconsistent evaluation settings, or unclear benchmark choices can make it difficult to judge whether reported improvements come from method changes or from evaluation choices (Ferrari Dacrema et al., 2019). Machine Learning is a particularly demanding case in this respect: it is methodologically central across disciplines, heavily dependent on shared and rapidly evolving artifacts, and reported through conventions that vary widely between communities and venues, which makes structured extraction both especially useful and especially difficult.

Naming variation and full-text distribution The relevant ML information is not expressed in one fixed form. The same artifact can be introduced by name, referred to through abbreviation, described by a generic phrase, or mentioned again in another section. This is especially important for ML-related artifacts, where model names, architectures, datasets, and generic mentions can require fine-grained distinctions (Otto et al., 2023). Important evidence about data use, training procedure, evaluation design, and comparisons is often distributed across full papers rather than concentrated in

abstracts (Luan et al., 2018; Jain et al., 2020). Systems that rely only on abstracts or short snippets therefore risk missing evidence needed for benchmark tracking, method comparison, and reproducibility-oriented analysis (Jain et al., 2020; Pineau et al., 2021).

From reading problem to scholarly IE These conditions turn the reading problem into a scholarly IE problem. Reliable extraction must identify typed mentions of ML-related artifacts and keep them linked to the textual evidence from which they were derived (Luan et al., 2018). Survey work on IE scientific articles shows that such methods are used to reduce manual effort and support structured access to scholarly content (Nasar et al., 2018). For this thesis, the relevant target is not only mention detection, but extraction of ML concepts, artifact mentions, and typed semantic relations from full-text publications, traceable to the supporting sentences. Recent generative approaches have shifted what is technically feasible for span- and record-level extraction, but the questions of evidence traceability, fine-grained typing, and cross-view comparability that motivate this thesis remain open regardless of the underlying extraction method.

Relations and workflow records Entity extraction alone is not sufficient for representing how ML concepts and artifacts are described, used, compared, and referenced in scholarly publications. Knowing that a paper mentions a model, a dataset, and a task does not show how these artifacts are used: whether the model is trained, evaluated, or only cited; whether the dataset is used for training or evaluation; or whether the task is introduced by the paper or treated as an established task in the field. Scientific ERE addresses this by modeling research concepts and typed relations between them (Luan et al., 2018). For ML-focused scholarly IE, such relations are needed to represent links between models, datasets, methods, tasks, benchmarks, properties, comparisons, and external references.

Generalization, comparability, and use Extraction results should therefore not be evaluated within a single dataset or annotation schema alone. Scientific IE resources differ in corpus selection, annotation scope, label granularity, and evaluation setup, but these differences are not merely surface inconsistencies to be normalized away. They are representational views over scholarly evidence, each shaped by the goals, prior commitments, and intended uses of the project that produced it. The same is true of any other reconstruction of the same papers, whether produced by a human reader, an annotation guideline, or a language model. Treating these differences as views rather than as noise opens a question that this thesis takes seriously: how methods, annotations, and extracted records developed under one view can be related to those developed under another. Extraction, generalization, and exploration are connected in this thesis by a common object of study: records of ML artifacts and their roles, traceable to their supporting text and studied across the views that produce and consume them.

1.2 Problem Statement and Research Goal

The central problem addressed in this thesis is that machine-learning artifact information is difficult to reliably extract, faithfully represent, and compare across scholarly

publications. Questions about training data, evaluation data, addressed tasks, reused models, and comparison baselines cannot be answered by mention detection alone. This difficulty arises from four connected challenges:
First, artifact mentions and experimental links are distributed across full texts and are not limited to abstracts. Important evidence can appear in method sections, experiment sections, captions, tables, appendices, and surrounding explanatory text. This makes abstract-only extraction insufficient for many questions about ML workflows.
Second, existing scholarly IE resources often use entity categories that are too coarse for ML-centered analysis. For example, a model architecture, a concrete trained model, a general method, a dataset, and a data source can play different roles in an experimental setup, while generic artifact references add a further distinction. If these distinctions are merged into broad labels, downstream analysis cannot reliably capture what was actually used, reused, or compared.
Third, many relevant statements in ML papers are relational. Entity extraction can identify mentions of models, datasets, tasks, and methods, but it does not by itself show how these artifacts are connected. Typed relations such as trained-on, evaluated-on, applied-to, benchmark-for, and compared-to are needed to represent experimental workflows as structured records. These records should remain traceable to the sentences from which they were derived.
Fourth, scientific ERE datasets differ in corpus selection, schema granularity, annotation density, and relation annotation criteria. These differences are best understood not only as inconsistencies, but also as distinct representational views over scholarly evidence. They make benchmark scores difficult to compare directly and make it unclear whether models learn reusable extraction behavior or view-specific annotation patterns. Schema alignment can support controlled comparison, but it can also remove distinctions that matter for relation extraction. Cross-dataset evaluation must therefore study both what can be aligned across views and what should remain view-specific. The research goal of this thesis is therefore to develop and evaluate methods for extracting evidence-linked records of machine-learning artifacts and relations from full-text publications, and for comparing these records across different annotation and extraction settings. The following research questions decompose this goal into extraction, cross-dataset robustness, and use-oriented analysis.
RQ-A (Extraction from full text): How can machine-learning artifacts and their workflow relations be extracted from full-text scholarly publications as fine-grained, evidence-linked records? Success is measured by entity and relation extraction quality under span- and label-based evaluation, by coverage of key artifact types and relation types, and by the ability to trace extracted records to supporting text spans.

RQ-A1 **Target granularity.** Which artifact distinctions are needed for ML papers (e.g., model architecture vs. trained model instance), and how can they be represented in an entity schema that remains learnable and evaluable?

RQ-A2 **Full-text evidence.** How can fine-grained artifact and relation annotations be created across full-text scholarly publications while keeping extracted records traceable to sentence-level textual evidence?

RQ-A3 **Workflow relations.** How can workflow relations between ML artifacts, such as trained-on, evaluated-on, and compared-to, be modeled and extracted as sentence-level relation evidence?

RQ-B (Robustness, comparability, and generalization): How can scientific entity and relation extraction models be compared and transferred across datasets when the datasets overlap conceptually but differ in schema design, annotation density, corpus selection, and relation semantics?

Success is measured by whether the thesis enables controlled comparison across scientific ERE datasets, evaluates transfer between resources that are related but not directly interchangeable, and clarifies when shared schemas support generalization and when dataset-specific schema distinctions must be preserved.

RQ-B1 **Controlled comparison.** How can scientific ERE datasets with different entity and relation schemas be aligned into a shared representation for controlled cross-dataset evaluation?

RQ-B2 **Transfer under unified schemas.** To what extent do models trained on one scientific ERE dataset transfer to another when predictions and gold annotations are mapped into a shared schema?

RQ-B3 **Schema-preserving generalization.** Can shared representations improve scientific ERE across benchmarks when each dataset keeps its own original prediction space?

RQ-C (Use and analysis at scale): How can extracted machine-learning knowledge be organized and presented so that users can inspect textual evidence and analyze artifact mentions across paper collections? Success is measured by an end-to-end workflow that applies extraction to paper collections, supports interactive inspection of evidence, and enables analysis tasks such as aggregation by artifact and comparison across papers.

RQ-C1 **Evidence and provenance.** Which record structure and evidence-linking strategy can support inspection and verification of extracted ML artifact mentions in practice?

RQ-C2 **Aggregation across papers.** How can extracted artifact mentions be grouped and summarized across papers, and where do mention-level aggregation and missing normalization limit comparison?

RQ-C3 **User tasks.** Which exploration tasks are directly supported by entity-level artifact extraction, and where are the limits of text-only and mention-level coverage?

1.3 Contributions of this Thesis

This cumulative thesis contributes five building blocks for extracting and using ML-related information from scholarly publications. As Figure 1.1 illustrates, they form a single line of work: fine-grained extraction and evidence linking feed cross-schema generalization, both rest on a set of shared design principles, and together they yield datasets, models, and a user-facing system. The five contributions are:

- **Fine-grained entity recognition for ML research (GSAP-NER).** The thesis introduces a fine-grained entity schema for ML artifacts and a manually annotated corpus of 100 full-text computer science publications, together with baseline NER models for these entity types.
- **LLM-assisted extraction of software information.** The thesis investigates large language model (LLM) based extraction for software mentions and their attributes, using task reformulation and retrieval-based prompting to support extraction in settings with limited supervision.
- **Modeling relations between machine learning artifacts (GSAP-ERE).** Building on the entity layer, the thesis introduces a fine-grained full-text ERE dataset that models key relations between ML artifacts (e.g., links between methods/models, datasets, and tasks), enabling structured representations of experimental workflows.
- **Generalizing scientific ERE across benchmarks without schema collapse.** The thesis aligns SciER and SciNLP with the introduced GSAP-ERE resource to study cross-dataset transfer across scientific ERE benchmarks (Q. Zhang et al., 2024; Duan et al., 2025). It shows that scientific ERE generalizes better when shared representations are learned without collapsing dataset-specific schemas.
- **Exploration and analysis of extracted machine learning knowledge (Methods Miner).** The thesis contributes a user-facing tool that applies the extraction models to scholarly corpora and supports interactive exploration of extracted ML artifacts with links back to textual evidence.

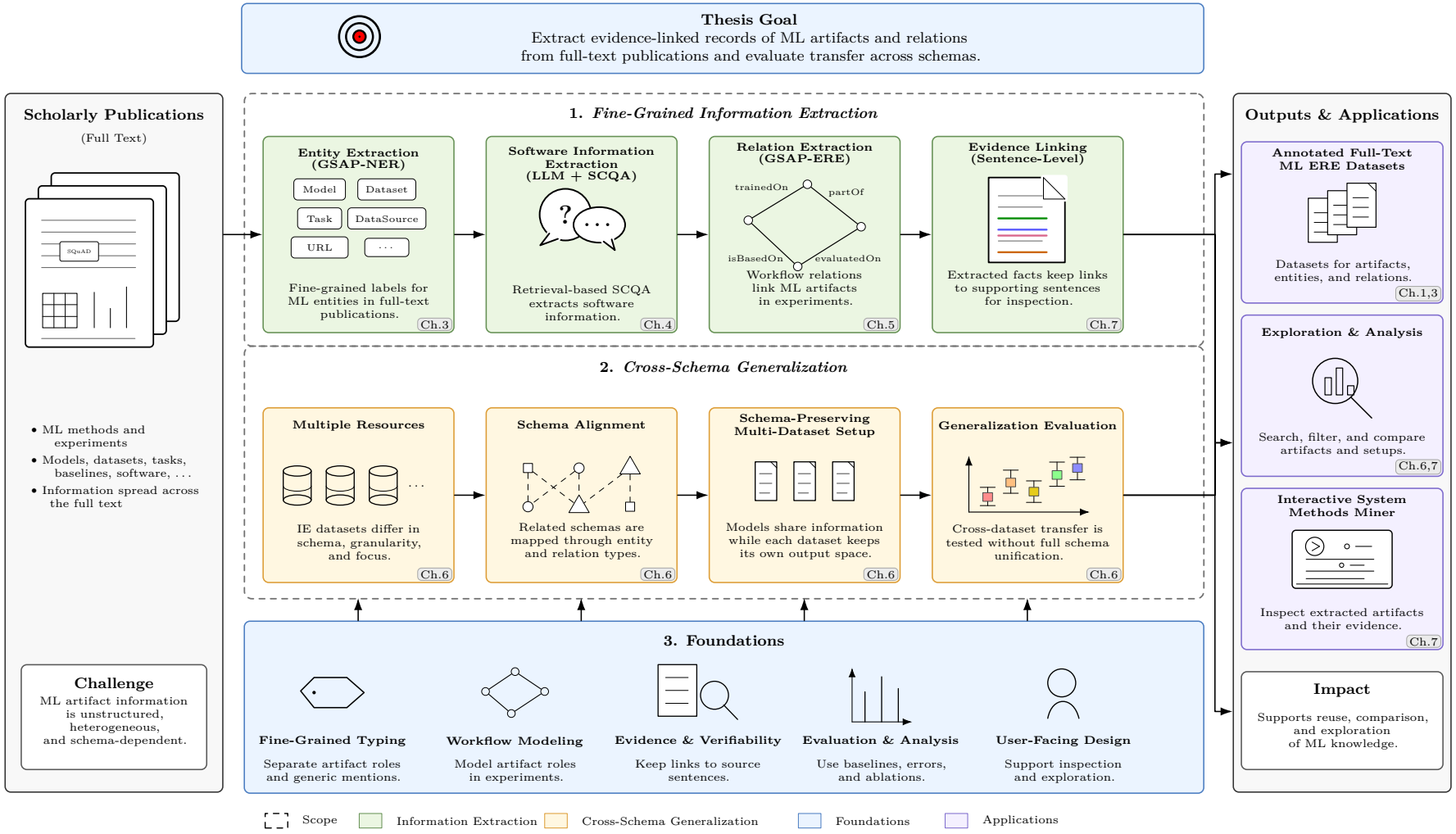


Figure 1.1: Overview of the thesis. The dissertation develops methods to extract fine-grained information about ML artifacts and their relations from full-text publications, links them to evidence, and generalizes across different schemas and resources. The results are made accessible through a user-centered exploration system to support analysis and reuse.

1.4 Contributing Publications

1.4.1 Main Contributing Articles[1]

GSAP-NER:
A Novel Task, Corpus, and Baseline for Scholarly Entity Extraction Focused on Machine Learning Models and Datasets
(Otto et al., 2023)

Authors: Wolfgang Otto, Matthäus Zloch, Lu Gan, Saurav Karmakar, and Stefan Dietze

In: *Findings of the Association for Computational Linguistics: EMNLP 2023.* Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Association for Computational Linguistics, Singapore, 2023, pp. 8166–8176

DOI: 10.18653/v1/2023.findings-emnlp.548

Status: Published Findings of ACL: EMNLP 2023 long paper (Archival, Peer-reviewed)

Role in this Thesis: This paper establishes the entity-level basis of the thesis by defining a fine-grained ML-focused entity inventory and by providing full-text annotations and baseline models. It shows how machine-learning artifacts can be represented as typed textual mentions rather than as one broad class of scientific entities. The resulting entity view is reused in GSAP-ERE, where ML artifacts are connected through workflow relations. It also supports the later cross-dataset chapter, where schema granularity becomes central, and the Methods Miner system, where extracted entities are inspected with sentence-level evidence. In the thesis line, this paper therefore provides the first tested representation of ML-related scholarly entities and prepares the move from entity extraction to relation-based workflow records.

Author Contribution Summary: I was first author and led the task conception, paper selection, experimental setup, baseline experiments, result analysis, manuscript coordination, and publication process. The entity schema, annotation guideline, and annotation process were refined in close collaboration with co-authors and annotators. I presented the poster at the EMNLP conference.

Summary: The paper introduces GSAP-NER, a fine-grained scholarly NER task for extracting ML-related entities from full-text publications. It separates artifact types that are often collapsed in earlier scholarly IE resources, including concrete ML models, model architectures, datasets, data sources, tasks, methods, and generic artifact mentions. The paper releases a manually annotated corpus of 100 full-text computer science publications and reports baseline experiments for the proposed entity inventory. The results show that supervised models can learn the main artifact types, while conceptual labels, informal mentions, and context-dependent distinctions remain difficult. These findings motivate the next thesis step, where extracted ML artifacts are connected through typed relations to represent experimental workflows.

[1] Detailed author contributions for the main contributing articles are provided in Appendix A.

Enhancing Software-Related Information Extraction via Single-Choice Question Answering with Large Language Models
(Otto et al., 2024)

Authors: Wolfgang Otto, Sharmila Upadhyaya, and Stefan Dietze

In: *Natural Scientific Language Processing and Research Knowledge Graphs. NSLP 2024.* Ed. by Georg Rehm et al. Springer Nature, Cham, 2024, pp. 289–306

DOI: 10.1007/978-3-031-65794-8_21

Status: Published NSLP 2024 workshop paper (Archival, Peer-Reviewed)

Role in this Thesis: This paper complements the thesis' supervised extraction line by investigating LLM-assisted extraction of software-related information under limited supervision. It adds a second methodological angle by reformulating extraction as prompted single-choice question answering with retrieval-based context. The paper is relevant to the thesis because software is another scholarly artifact needed for reuse, reproducibility, and interpretation of experimental work. Its mentions and attributes show similar problems as ML models and datasets, including naming variation, local evidence, and schema-specific relation decisions. In the thesis line, the paper helps position LLM-based prompting as a useful controlled baseline and as a contrast to fine-tuned extraction models. It therefore supports the later discussion of method trade-offs, especially where flexible prompting helps but strict schema-bound extraction and evaluation remain necessary.

Author Contribution Summary: I was first author and took primary responsibility for the paper from the SOMD shared-task participation to the final manuscript. I led the methodological approach, designed the single-choice QA and retrieval-based prompting setup, implemented the experimental pipeline, ran the experiments, analyzed the results, and wrote the manuscript. The co-authors reviewed drafts, suggested revisions, and helped improve the presentation, positioning, and final form of the paper. I presented the poster at the NSLP workshop.

Summary: The paper reports the system submitted to the SOMD shared task, where the goal is to extract software mentions, attributes, and relations from scholarly text. It combines in-context learning, retrieval-based example selection, and a single-choice QA formulation for relation extraction. This setup turns relation extraction into a constrained decision problem, making the output easier to compare against the shared-task labels. The results show that task reformulation can provide a practical LLM-based baseline for software-focused scholarly IE in limited-supervision settings. At the same time, the paper shows why prompt-based extraction still needs controlled schemas, careful example selection, and strict evaluation. For the thesis, it provides a methodological contrast to the supervised extraction chapters and prepares the later discussion of robustness and evidence-oriented extraction across different extraction settings.

GSAP-ERE: Fine-Grained Scholarly Entity and Relation Extraction Focused on Machine Learning

(Otto et al., 2026)

Authors: Wolfgang Otto, Lu Gan, Sharmila Upadhyaya, Saurav Karmakar, Stefan Dietze

In: *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(38), 32600–32609

DOI: 10.1609/aaai.v40i38.40537

Status: Published AAAI 2026 Main Technical Track paper (Archival, Peer-reviewed)

Role in this Thesis: This paper extends the thesis from entity extraction to entity-and-relation extraction by introducing typed relations between ML-related artifacts. It builds directly on the GSAP-NER entity view and turns lists of extracted artifact mentions into structured workflow records. This is central for the thesis because ML papers do not only mention models, datasets, tasks, and methods, but also describe how these artifacts are trained, evaluated, applied, compared, and referenced. The paper therefore provides the main resource and experimental basis for answering how ML workflows can be extracted from full-text scholarly publications. It also prepares the next thesis step: the cross-dataset chapter uses GSAP-ERE as one of the main scientific ERE resources for studying schema alignment, transfer, and schema-preserving generalization. In the thesis line, GSAP-ERE is the bridge between fine-grained extraction and later questions of robustness, comparability, and evidence-based exploration.

Author Contribution Summary: I was first author and led the overall task conception, dataset design, study design, supervised modeling approaches, result analysis, manuscript coordination, and publication process. The relation inventory, annotation guideline, and annotation process were developed collaboratively, while I created the seven semantic relation groups, prepared the data, managed the technical annotation workflow, and contributed to the interpretation of the LLM baseline. I presented the paper at the conference and presented the poster at AAAI together with the third author.

Summary: The paper introduces GSAP-ERE, a fine-grained full-text dataset for entity and relation extraction focused on machine-learning publications. It extends the earlier entity-only view with relation types that connect ML artifacts, including models, methods, datasets, tasks, properties, comparisons, and references. The dataset contains 10 entity types and 18 relation types annotated in 100 full-text ML publications, providing a resource for modeling experimental workflows at sentence level. The paper reports supervised baselines and compares them with prompting-based LLM setups. The results show that relation extraction is necessary for moving beyond artifact lists, but also that fine-grained scholarly ERE remains difficult, especially for LLM prompting under strict evaluation. For the thesis, GSAP-ERE provides the workflow-level extraction layer and sets up the following chapter, where related scientific ERE datasets are aligned and compared without collapsing their original schema distinctions.

Generalizing Without Collapsing: Schema-Preserving Transfer Across Scientific ERE Benchmarks

Authors: Wolfgang Otto, Stefan Dietze

Status: Submitted EACL 2027 long paper (Not yet accepted or published.)

Role in this Thesis: This paper provides the thesis-level generalization layer for scientific entity and relation extraction. It moves beyond evaluating extraction models on single datasets and asks whether models learn reusable extraction behavior across scientific ERE benchmarks. The paper builds directly on the earlier extraction chapters: GSAP-NER defines the fine-grained entity layer, and GSAP-ERE extends this layer to workflow relations between ML artifacts. By comparing and aligning SciER, SciNLP, and GSAP-ERE, the paper turns these resources into a test case for robustness, comparability, and cross-dataset generalization. It shows that a shared schema enables controlled comparison, but is not sufficient for robust relation extraction transfer. The paper therefore adds a schema-preserving transfer perspective to the thesis by training shared representations while keeping each dataset's original prediction space. This also prepares the following application-oriented chapter, because it clarifies which extracted signals can be treated as reusable across resources and which remain tied to dataset-specific schemas. In this way, the paper forms the bridge from fine-grained extraction to evidence-based exploration of extracted ML knowledge.

Author Contribution Summary: The second author contributed through intensive methodological discussion, iterative development of the contextual setting, result interpretation, feedback on the argument and method, and manuscript revisions.

Summary: The paper studies whether scientific ERE models generalize across SciER, SciNLP, and GSAP-ERE, three full-text scientific ERE datasets with overlapping targets but different schemas, corpus distributions, and annotation practices. It introduces an alignment pipeline for controlled cross-dataset evaluation and uses a unified label space to test transfer under shared labels. The results show that schema unification enables direct comparison, but does not provide robust transfer for relation extraction, because similarly named relation labels can encode different annotation semantics. The paper therefore evaluates a dataset-conditioned multi-head ERE model that shares the encoder and relation representations across datasets while preserving separate NER and RE heads for each original schema. This schema-preserving setup improves performance on SciER and SciNLP, especially for relation extraction, while GSAP-ERE benefits less from adding smaller or less fine-grained sources. The paper also releases the unified benchmark, schema alignment pipeline, model outputs, code, and an inspection application for analyzing corpus composition, predictions, and cross-dataset comparisons.

Methods Miner: Exploring Machine Learning Methods, Tasks, and Datasets in Scholarly Publications

Authors: Wolfgang Otto, Sharmila Upadhyaya, Lu Gan, Kanishka Silva, Stefan Dietze

Status: To be submitted as demo paper (Not yet accepted or published.)

Role in this Thesis: This paper demonstrates the practical use of the thesis contributions by making extracted ML-related information available in an interactive exploration workflow. It shifts the thesis from dataset construction and model evaluation to the question of how evidence-linked extraction results can support users working with focused paper collections. The paper builds on the earlier entity-level work, especially GSAP-NER, by using fine-grained ML artifact extraction as the basis for browsing publications, entities, and textual contexts. It is relevant to the thesis because extracted artifacts are only useful for analysis when users can inspect the evidence from which they were derived. Methods Miner therefore makes the thesis's evidence-oriented view visible at the system level: paper-level overviews remain connected to sentence-level contexts. The paper also shows the practical limits of the current extraction setting, especially where entity mentions still need normalization, linking, relation-aware grouping, and stronger support for tables, figures, and cross-sentence evidence. In the thesis line, this paper provides the application-oriented endpoint: it turns the earlier extraction components into a user-facing tool and prepares the final discussion of usefulness, limitations, and future work.

Author Contribution Summary: I was first author, managed the writing process, planned the system workflow, implemented major parts of the prototype, and integrated the processing pipeline into the final system. The main conceptualization was developed jointly by the co-authors, and co-authors contributed special functionalities, the initial setup of selected services, feedback on the demonstration scenario, and manuscript revisions.

Summary: The paper presents Methods Miner, a web-based system for exploring ML-related artifacts in user-selected scholarly publications. Users can upload PDF papers and receive an overview of extracted entities such as methods, tasks, datasets, models, and related artifact mentions. The system keeps extracted mentions linked to sentence-level contexts, so users can inspect the textual evidence behind each extracted item. The backend combines PDF-to-text conversion with Grobid, fine-grained scholarly NER based on GSAP-NER, structured storage, and an API for serving publications, entities, and contexts. The frontend organizes this information into views for uploaded publications, extracted entity groups, and supporting text passages. This allows users to move between corpus-level overviews and local evidence in individual papers. For the thesis, the paper shows how the extraction resources and models developed in the earlier chapters can support interactive analysis. At the same time, it marks the current boundary of the approach, because reliable comparison across papers still requires stronger normalization, relation extraction, and support for evidence beyond plain text.

1.4.2 Other First-authored Publications and their Relation to this Thesis

Knowledge Extraction from Scholarly Publications:
The GESIS Contribution to the Rich Context Competition
(Otto et al., 2020)

Authors: Wolfgang Otto, Andrea Zielinski, Behnam Ghavimi, Dimitar Dimitrov, Narges Tavakolpoursaleh, Karam Abdulahhad, Katarina Boland, Stefan Dietze.

In: *Rich Search and Discovery for Research Datasets*. Ed. by Julia Lane, Ian Mulvany, and Paco Nathan. Sage, Los Angeles u.a., 2020, pp. 107–127

URL: `https://study.sagepub.com/richcontext`

Status: Published book chapter (Peer reviewed)

Note: Second Round of Rich Context Competition, invitation to workshop at New York University as one of three teams.

Connection to this Thesis: This work is an early step toward the thesis focus on extracting structured information from scholarly publications. It is particularly connected to the thesis through its dataset detection component, which addresses the identification and linking of dataset mentions in full-text publications. While the later thesis work focuses on machine-learning-related artifacts, this contribution already studies how dataset mentions can be detected in scholarly text and connected to structured reference information.

Author's Contribution: First author. I led the GESIS contribution to the Rich Context Competition and implemented major parts of the extraction pipeline, with a particular focus on dataset detection. The research field classification component was developed jointly by the third author and me. The research method extraction component was developed by Andrea Zielinski and integrated into the overall extraction pipeline by me. I presented the system and results at the competition workshop and was primarily responsible for the manuscript, with contributions and revisions from co-authors.

Summary: The chapter presents the GESIS contribution to the Rich Context Competition, which addressed the extraction of structured information from scholarly publications. The system follows a module-based pipeline that preprocesses PDF documents, extracts dataset mentions, links them to known dataset references where possible, identifies research method mentions, and classifies publications by research field. Dataset extraction is treated as a named entity recognition and linking task, while research method extraction combines named entity recognition, linking, lexical resources, and word embeddings. Research field classification is implemented as a supervised multi-label classification task trained on social science publication metadata.

Highly Cited References in PLOS ONE
and their In-Text Usage over Time
(Otto et al., 2019)

Authors: Wolfgang Otto, Behnam Ghavimi, Philipp Mayr, Rajesh Piryani, Vivek Kumar Singh

In: *Proceedings of the 17th International Conference on Scientometrics & Informetrics (ISSI 2019)*. Ed. by Giuseppe Catalano, Cinzia Daraio, Martina Gregori, Henk F. Moed, and Giancarlo Ruocco. Rome: Edizioni Efesto, 2019, pp. 1531–1536

URL: `https://www.issi-society.org/proceedings/issi_2019/ISSI%202019%20-%20Proceedings%20VOLUME%20II.pdf`

Status: Published ISSI research-in-progress paper

Connection to this Thesis: This work is connected to the thesis through its focus on extracting structured information from scholarly publications. It studies citation contexts in full-text articles and uses section information, citation age, and co-citation patterns to describe how highly cited references are used over time. While the thesis focuses on machine-learning-related artifacts such as models, datasets, and tasks, this paper addresses a related scholarly IE setting in which citation contexts are used as evidence for literature-level analysis.

Author's Contribution: Author. I led the research, coordinated the analysis, and wrote the paper with contributions and revisions from co-authors. The paper was not presented by me at the ISSI workshop.

Summary: The paper analyzes highly cited references in a full-text corpus of PLOS ONE articles published between 2006 and 2017. It selects 666 PubMed-indexed references cited in more than 100 PLOS ONE papers and studies 173,630 citation contexts from 62,127 citing papers. For each citation context, the analysis considers the IMRaD section, the citation interval between the citing and cited publication, and the number of co-cited references in the same sentence. The results show that highly cited references differ in their section-level citation patterns and that older references are increasingly cited in method sections. The paper also finds that the apparent decline in co-citation counts over longer citation intervals is explained by the growing share of method-section citations, where citation contexts contain fewer co-cited references.

Team GESIS Cologne:
An All in All Sentence-Based Approach for FEVER
(Otto, 2018)

Author: Wolfgang Otto

In: *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER).* Ed. by James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 145–149

DOI: 10.18653/v1/W18-5524

Status: Published workshop paper

Note: Shared Task Rank 8/24

Connection to this Thesis: This work is connected to the thesis through its focus on extracting and using textual evidence from large encyclopedic text collections. It addresses the retrieval of relevant evidence sentences and the classification of their relation to a given claim. Although the task is fact verification rather than scholarly IE, the paper contributes an early sentence-level pipeline design for retrieving, ranking, and interpreting evidence-bearing text.

Author's Contribution: Sole author. I developed the complete FEVER shared-task pipeline, including the sentence retrieval component, the sentence-wise entailment setup, and the ensemble-based aggregation step. I ran the experiments, submitted the system to the shared task, and wrote the workshop paper.

Summary: The paper presents a sentence-based system for the FEVER shared task. The pipeline retrieves candidate evidence sentences directly from Wikipedia introduction sentences, without a separate document retrieval step. For each claim–sentence pair, a decomposable attention model predicts whether the sentence supports, refutes, partly supports, partly refutes, or does not provide enough information for the claim. The predictions for the top retrieved sentences are then combined with a random forest classifier to produce the final claim-level decision. On the shared-task test set, the system achieved a FEVER score of 40.77 and ranked eighth.

1.4.3 Related Co-authored Articles

Sharmila Upadhyaya, Wolfgang Otto, Frank Krüger, and Stefan Dietze (2025). **"SOMD2025: A Challenging Shared Task for Software Related Information Extraction"**. In: *Proceedings of the Fifth Workshop on Scholarly Document Processing (SDP 2025)*, pp. 137–145 (Upadhyaya et al., 2025)

Rajesh Piryani, Wolfgang Otto, Philipp Mayr, and Vivek Kumar Singh (2019). **"Analysing author name mentions in citation contexts of highly cited publications"**. In: *Proceedings of the 4th Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL 2019)*. Ed. by Muthu Kumar Chandrasekaran and Philipp Mayr. RWTH, Aachen, pp. 145–152. (Piryani et al., 2019)

Behnam Ghavimi, Wolfgang Otto, and Philipp Mayr. **"An evaluation of the effect of reference strings and segmentation on citation matching"**. In: *Digital Libraries for Open Knowledge : 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019*, Oslo, Norway, September 9-12, 2019, Proceedings. Ed. by Antoine Doucet et al. Springer, Cham, 2019, pp. 365–369. (Ghavimi et al., 2019)

1.5 Outline

The preceding section documents the publications included in this cumulative thesis. Before outlining the chapter sequence, Table 1.1 reconnects the paper-based chapters to the research questions and shows how they build the cumulative thesis line. The

Chapter	RQ focus	Core output	Role in thesis line
Ch. 3	RQ-A1/A2: artifact granularity and full-text evidence	GSAP-NER: fine-grained ML artifact schema, full-text NER corpus, and baseline models.	Entity layer for relation extraction, schema comparison, and evidence-based exploration.
Ch. 4	RQ-A: extraction under limited supervision	LLM-assisted extraction of software mentions, attributes, and relations using single-choice QA and retrieval.	Complementary extraction strategy for scholarly artifact extraction.
Ch. 5	RQ-A2/A3: full-text evidence and workflow relations	GSAP-ERE: full-text ERE corpus with ML artifact entities, workflow relations, and baseline models.	Relation layer for workflow-level records and cross-dataset comparison.
Ch. 6	RQ-B1–B3: schema alignment, transfer, and schema preservation	Schema alignment, cross-dataset evaluation, and schema-preserving transfer across scientific ERE benchmarks.	Robustness and comparability across representational views.
Ch. 7	RQ-C1–C3: evidence, aggregation, and user tasks	Methods Miner: exploration system for extracted ML artifacts with sentence-level evidence.	Use-oriented analysis and inspection of extracted records.

Table 1.1: Overview of the thesis line across chapters, research questions, and cumulative contributions.

included papers form the basis of the paper chapters and contribute complementary building blocks that depend on each other cumulatively.

Chapter 1 introduces the motivation, problem statement, and research goal of the thesis, defines the three main research questions (RQ-A to RQ-C), summarizes the main contributions, and explains how the included publications connect.

Chapter 2 reviews related work in scholarly information extraction and provides shared background used across RQ-A to RQ-C, with a focus on (i) entity recognition, relation extraction, and joint ERE methods, including supervised and LLM-based approaches, (ii) dataset scope decisions (abstract-level vs. full-text, coarse-grained vs. fine-grained), and (iii) prior work on extracting machine-learning-related artifacts.

Chapter 3 (*Fine-Grained Entity Recognition for Machine Learning Research*) addresses RQ-A by presenting GSAP-NER, a fine-grained entity schema and a manually annotated full-text corpus for ML artifacts, and by reporting baseline models and analysis of full-text annotation and modeling challenges.

Chapter 4 (*LLM-assisted Extraction of Software Information*) addresses RQ-A and

contributes to RQ-B by investigating LLM-based extraction for software-related information, reformulating extraction as single-choice question answering, and evaluating retrieval-based in-context learning strategies under limited supervision.
Chapter 5 (*Modeling Relations between Machine Learning Artifacts*) addresses RQ-A by introducing GSAP-ERE, defining relation types that capture key ML workflow links, and reporting baseline results for extracting these relations from full text.
Chapter 6 (*Schema-Preserving Generalization Across Scientific ERE Benchmarks*) addresses RQ-B by analyzing scope and schema differences, introducing a canonical schema with mapping-based alignment, and enabling controlled cross-dataset evaluation and transfer analysis under unified settings.
Chapter 7 (*Exploration and Analysis of Extracted Machine Learning Knowledge*) addresses RQ-C by presenting Methods Miner, a web-based system that applies the extraction components to paper collections and supports interactive exploration with links back to textual evidence.
Finally, Chapter 8 synthesizes findings across RQ-A to RQ-C, summarizes the thesis contributions, discusses limitations, and outlines directions for future work.

2

Background: Information Extraction, Scientific NLP, and Scholarly IE

This chapter provides the shared background for the thesis and explains how the included publications are positioned within scholarly IE. It does not repeat the detailed related work of the individual paper chapters. Instead, it introduces the concepts, task settings, resource types, modeling choices, and evaluation problems that are needed to understand the cumulative thesis line.

The role of this chapter is threefold. First, it defines the information extraction tasks that support the extraction research question of the thesis, especially named entity recognition, relation extraction, and joint entity and relation extraction. Second, it explains why scientific publications require a domain-specific view, since scholarly evidence is distributed across document sections, shaped by technical terminology, and often tied to datasets, models, software, tasks, and evaluation settings. Third, it prepares the later discussion of robustness and use by introducing schema mismatch, cross-dataset comparability, evidence links, and exploration systems as recurring thesis-level issues.

The chapter is organized from general task definitions to downstream use. Section 2.1 introduces the main IE tasks and extraction settings. Section 2.2 narrows the focus to scientific NLP and scholarly IE. Section 2.3 defines the thesis-relevant extraction targets, including ML artifacts, mentions, workflow relations, and evidence-linked records. Section 2.4 discusses corpora and benchmarks as designed resources that shape what can be learned and compared. Section 2.5 reviews supervised, LLM-based, and hybrid modeling approaches. Section 2.6 explains why annotation rules, matching criteria, and schema differences are central for evaluation and transfer. Finally, Section 2.7 connects extraction outputs to exploration systems, where extracted records must remain searchable, aggregable, and linked to their textual evidence.

2.1 Information Extraction Tasks and Settings

IE refers to a family of NLP tasks that identify relevant information in text and represent it as structured outputs such as entity spans, relation tuples, event structures, slot fillers, or records (Cowie and Lehnert, 1996). For this thesis, IE provides the methodological basis for extracting machine-learning artifacts and their relations from scholarly publications. This section introduces the task vocabulary and design choices needed across the paper chapters, while the paper-specific chapters discuss their closest prior work in more detail.

The most relevant IE tasks for this thesis are named entity recognition, relation

extraction, and joint entity–relation extraction. Named entity recognition identifies spans in text and assigns them to semantic types. This task is central to Chapter 3, where models, model architectures, datasets, data sources, tasks, and generic artifact mentions are treated as fine-grained entity types. NER is often studied through shared benchmark settings with fixed type inventories and controlled evaluation, as in the CoNLL-2003 shared task (Tjong Kim Sang and De Meulder, 2003). Relation extraction predicts typed links between entity mentions, for example between two entities that stand in a semantic or task-specific relation. Early work on relation extraction already treated relation prediction as a supervised classification problem over entity pairs and their syntactic or contextual evidence (Zelenko et al., 2002). This task is central to Chapter 5, where relations describe how machine-learning artifacts are used in experimental workflows. Joint entity–relation extraction models entity spans and relation labels in a shared setup rather than treating them as fully separate steps (Q. Li and Ji, 2014; Miwa and Bansal, 2016). Pipeline systems first detect entity mentions and then classify candidate pairs, which makes errors in entity detection affect later relation decisions. Joint models can reduce this dependency, but they depend strongly on consistent span boundaries, type definitions, and relation labels. Event extraction and slot filling are related IE task families that also produce structured representations from text. Event extraction usually identifies event triggers and assigns argument roles to participating entities (Simon et al., 2024). Slot filling usually fills predefined fields in a record or template, a formulation that goes back to early IE evaluations and remains active in domain-specific IE (Grishman, 1996; Papanikolaou et al., 2022). These task families are useful background because they show how IE can move from local spans to structured records, but the main modeling line of this thesis remains entity and relation extraction.

A second central design choice concerns the annotation unit, the coverage of textual evidence, and the level at which extraction is evaluated. Many earlier scientific IE resources focus on abstracts, selected paragraphs, or selected sentences, which makes annotation and evaluation easier but limits the kinds of evidence that can be observed (Augenstein et al., 2017; Gábor et al., 2018; Luan et al., 2018; Hou et al., 2021; Heddes et al., 2021). Full-text scientific IE resources such as SciREX and SciER address this limitation by treating scientific extraction as a document-level or full-paper task (Jain et al., 2020; Q. Zhang et al., 2024).

In machine-learning papers, important information about datasets, models, training setups, evaluation settings, baselines, and comparisons is often distributed across the full paper (Jain et al., 2020; Yousuf et al., 2022). This does not mean that every extraction decision must be modeled with full-document input. A system may still operate on sentences or local contexts. However, the corpus should contain full-text evidence, and evaluation should be organized so that performance can be interpreted at the level at which downstream use occurs. SciREX makes this issue explicit by formulating scientific IE as a document-level task over full papers (Jain et al., 2020). SciER continues this line for full-text scientific entity and relation extraction over datasets, methods, and tasks (Q. Zhang et al., 2024). For this thesis, full-text annotated resources and paper-level test splits are important even when the extraction model predicts sentence-level entities or relations. This point connects directly to Chapters 3 and 5, where full-text corpora are annotated, and to Chapter 6, where dataset comparison and transfer are evaluated across papers rather than only isolated sentences. It also matters for Chapter 7, because

user-facing exploration is not a sentence-level task in practice. Users inspect papers, compare artifacts across papers, and expect extracted records to remain linked to the source evidence.

A third design choice concerns whether the extraction schema is closed or open. Closed-schema IE uses a predefined label set and supports controlled training and evaluation. Open IE relaxes this assumption by extracting relational statements without requiring relation-specific schemas or labeled examples in advance (Banko and Etzioni, 2008; Niklaus et al., 2018). This thesis mainly follows a closed-schema setting, in which predefined entity and relation type inventories are used to build the thesis corpora and to support the later cross-dataset evaluation. Open-schema IE can discover or generate labels more flexibly, but it makes evaluation and comparison less direct (Niklaus et al., 2018). The closed-schema choice is especially relevant for Chapter 6, where the same scientific evidence is compared through different dataset schemas and prediction spaces. The problem is therefore not only whether a model can extract a relation, but also which representational view defines the relation in the first place. This is why schema design, schema alignment, and schema-preserving evaluation become central later in the thesis.

IE research can also be organized by the trade-off between annotation quality and extraction scale. One stream is benchmark-driven IE, where manually curated datasets define a focused task, a fixed schema, and a controlled evaluation protocol. The strength of this stream is that gold labels support direct model comparison, detailed error analysis, and reproducible evaluation, while its limitation is that schemas and corpora are often narrow. A second stream targets larger-scale extraction and often replaces full manual annotation with weak or distant supervision. Distant supervision aligns existing structured facts to text mentions and uses these alignments as noisy labels for relation extraction (Mintz et al., 2009). The strength of this stream is broader corpus coverage and easier integration into knowledge bases or search systems, while its limitation is weaker label control, noisier evaluation, and harder error attribution. This thesis is positioned mainly on the benchmark-driven side, because it introduces and evaluates controlled schemas for fine-grained scholarly IE, but it also connects to scalable extraction through cross-dataset evaluation and later use in an exploration system.

A related shift concerns the models used for extraction: recent IE systems increasingly use LLMs and generative models, which can operate with prompts, instructions, or schema descriptions instead of task-specific classifiers. LLM-based IE changes how extraction can be implemented, but it does not remove the distinction between task definition, supervision, and evaluation.

Recent generative IE work frames extraction as producing structured outputs from instructions, prompts, or schemas (Lu et al., 2022; Z. Zhang et al., 2025). When prompts or instruction-tuned models use fixed labels and structured outputs, LLM-based IE follows the logic of closed benchmark-style extraction. This also appears in scientific-domain LLM extraction, where predefined entity and relation schemas are used to produce structured records and evaluate them with benchmark-style metrics (Dagdelen et al., 2024).

When prompts or instruction-tuned models use fixed labels and structured outputs, LLM-based IE follows the logic of closed benchmark-style extraction (Lu et al., 2022). At the same time, large schema-based instruction corpora such as IEPile show how

existing IE resources can be repackaged for broader LLM-based extraction across tasks and schemas (Gui et al., 2024). In scientific-domain extraction, predefined entity and relation schemas are also used to produce structured records and evaluate them with benchmark-style metrics (Dagdelen et al., 2024). Chapter 4 is connected to this point because it studies LLM-assisted extraction under limited supervision. In the thesis, LLM-based extraction is therefore treated as a complementary extraction strategy rather than as a replacement for controlled, fine-grained evaluation.

These task distinctions prepare the move to scholarly IE. The next section narrows the discussion to scientific text, where long documents, technical terminology, citation practices, full-text evidence, and scarce full-text ERE corpora make the general IE choices more difficult.

2.2 Scholarly Information Extraction and Scientific NLP

Scientific NLP studies NLP methods for scientific and scholarly text. A central reason for treating scientific text as a specific NLP setting is that scientific publications differ from general-domain documents in terminology, discourse structure, citation practices, and document organization. Domain-adapted models such as SciBERT show that pretraining on scientific publications can improve performance across scientific NLP tasks, which supports the view that scientific text requires specialized treatment (Beltagy et al., 2019). At the corpus level, resources such as S2ORC also show that scholarly publications are commonly represented together with metadata, abstracts, references, structured full text, citation mentions, figures, and tables (Lo et al., 2020). Scholarly IE is the part of this scientific NLP concerned with turning textual evidence from scholarly documents into structured entities, relations, records, or other machine-readable representations.

Scientific papers are not only long texts. They are structured documents in which evidence is distributed across titles, abstracts, sections, captions, references, tables, and figures. This structure affects extraction because entity mentions and relation evidence can depend on section context, citation context, repeated mentions, and the local role of a sentence in the paper. Work on structured scholarly corpora and citation-aware document processing makes this explicit by representing full texts together with sections, citation links, figures, and other document objects (Lo et al., 2020; Rohatgi et al., 2023). Citation context is one example of such document structure: citation intent classification uses surrounding text and structural information to distinguish whether a cited work is used for background, methods, or comparison (Cohan et al., 2019). For scholarly IE, these properties matter because the same surface form can have different roles depending on where and how it appears in the paper.

Scientific IE includes several task families. Some tasks extract keyphrases, entities, relations, and coreference links from scientific text. SciERC is a central example because it jointly models scientific entities, relations, and coreference for scientific knowledge graph construction (Luan et al., 2018). ScienceIE provides an earlier shared-task framing for extracting keyphrases and relations from scientific publications (Augenstein et al., 2017). Other scholarly-document tasks extract metadata, section structure, bibliographic information, tables, figures, captions, or citation information. These tasks are related to scholarly IE, but they often require different input units, parsing

assumptions, and evaluation methods.

For this thesis, the most important scope distinction is between abstract-level extraction and full-text extraction. Abstract-level resources are useful because they reduce annotation cost and simplify evaluation, but they can miss evidence that appears only in methods, experiments, appendices, captions, tables, or surrounding explanatory text. Full-text resources are therefore important for judging extraction methods in settings closer to downstream use. SciREX formulates scientific IE as a document-level task over full papers and includes salient entity identification and document-level relation identification (Jain et al., 2020). SciER continues this direction with full-text scientific entity and relation annotations for datasets, methods, and tasks (Q. Zhang et al., 2024). This point is central for the thesis because Chapters 3 and 5 use full-text corpora, Chapter 6 compares scientific ERE resources across papers, and Chapter 7 presents extracted records as paper-level evidence for user inspection.

The thesis focuses on text-based extraction of artifact mentions and semantic relations from full-text scholarly publications. This excludes some adjacent scholarly-document processing problems as main contributions. Metadata extraction, table structure extraction, figure and caption extraction, bibliography parsing, and document layout processing are important for complete scholarly-document understanding, but they require different annotation units and often different model architectures. Systems such as OCR++ illustrate this broader document-processing setting by extracting metadata, section structure, table and figure headings, URLs, footnotes, citations, and references from scholarly articles (M. Singh et al., 2016). In contrast, this thesis focuses on the text-level evidence needed to identify machine-learning artifacts and the relations that connect them in experimental workflows.

This boundary does not mean that adjacent evidence sources are irrelevant. Tables, figures, captions, and metadata can contain important information about methods, datasets, tasks, and results. They are treated here as neighboring directions that help define the limits of text-based extraction. Within this boundary, the next section turns to the extraction targets that are central for the thesis: artifact mentions, identifiers, workflow relations, and evidence-linked records.

2.3 Extraction Targets in Scholarly Publications

After defining IE as a task family and scholarly IE as the domain setting, this section specifies what the thesis treats as extractable scholarly knowledge. Scientific IE has commonly represented scientific knowledge through entities, relations, and coreference structures that can support later knowledge graph construction (Luan et al., 2018). For this thesis, the central extraction targets are machine-learning concepts, research artifacts, textual mentions, workflow relations, and evidence-linked records. This target view connects the entity-level extraction in Chapter 3, the software-focused extraction in Chapter 4, the relation extraction in Chapter 5, the cross-dataset comparison in Chapter 6, and the evidence-oriented exploration in Chapter 7.

A useful distinction for this thesis is between concepts, artifacts, and mentions. Concepts are abstract research objects or categories, such as tasks, methods, model architectures, problem formulations, or evaluation settings. Artifacts are reusable or referable research objects, such as datasets, pretrained models, software packages, benchmarks, data sources, or implemented systems. Mentions are the textual expres-

sions through which concepts and artifacts appear in papers, including names, aliases, abbreviations, descriptive phrases, and generic references. Scientific IE resources already differ in how they draw these boundaries: SciERC models scientific entities and relations at a broad level, while later resources such as SciDMT focus on large-scale mention detection for datasets, methods, and tasks (Luan et al., 2018; Pan et al., 2024). The thesis builds on this line but uses a more ML-specific target view.

Machine-learning publications require fine-grained artifact distinctions because broad scientific IE labels often collapse roles that matter for interpreting experiments. A model architecture, a concrete trained model, a general method, a dataset, a data source, a task, a benchmark, and a software tool may all be relevant, but they do not play the same role in a paper. SciREX and SciER show that methods, datasets, metrics, and tasks are important extraction targets in scientific papers, especially when extraction moves beyond isolated abstracts toward document-level or full-text settings (Jain et al., 2020; Q. Zhang et al., 2024). For ML-centered scholarly IE, these categories must be refined further so that artifact mentions can support later comparison, relation extraction, and paper-level analysis.

Datasets and software illustrate why research artifacts need to be treated as first-class extraction targets. Dataset mentions vary across and within publications, and extraction may need to connect surface forms to canonical dataset references (Prasad et al., 2019). Software is similarly important because software tools, libraries, and packages are often part of the reported research process but are not always represented clearly in bibliographic metadata (Du et al., 2021). This thesis therefore treats datasets, models, methods, software, and related objects not only as words to identify, but as research artifacts that may need to be grouped, compared, and inspected across papers. Detailed normalization and grounding are discussed later; here the important point is that extraction targets must be defined in a way that can support such later use.

Extraction targets are also representational choices. Different resources may describe overlapping scholarly evidence through different labels, scopes, and relation definitions. For example, one schema may distinguish datasets, methods, tasks, and metrics, while another may merge some of these categories or define relations at a different level of detail. Such choices determine what can later be aligned across datasets, what is lost during unification, and which distinctions must remain dataset-specific. This is central for the thesis because Chapter 6 treats scientific ERE resources not only as collections of labels, but as different views over similar kinds of scholarly evidence.

Entity mentions alone are not sufficient for reconstructing machine-learning experiments. A paper-level record should not only state that a model, dataset, task, or metric was mentioned. It should also represent how these artifacts are connected: which model was trained on which data, which dataset was used for evaluation, which task was addressed, which method was compared with which baseline, and which metric supported the reported comparison. Document-level resources such as SciREX make this relation-oriented view explicit by modeling salient entities and n-ary relations over scientific articles (Jain et al., 2020). In this thesis, workflow relations provide the bridge from artifact mentions to structured records of experimental setups.

The extracted records should remain linked to textual evidence. This is important for evaluation, error analysis, and user-facing inspection. A predicted artifact or relation is more useful when the sentence or passage supporting it can be inspected, because users can verify whether the extracted record is justified by the paper text. Evidence

links also matter for comparing datasets and schemas, since disagreements can arise not only from model errors but also from different decisions about which textual evidence counts as an instance of a target. For downstream use, evidence-linked records therefore connect extraction quality to practical interpretability.

Results, comparisons, metrics, and scientific claims form neighboring extraction targets. Work on result extraction from machine-learning papers shows that leaderboard-style information and reported scores can be extracted as structured records (Kardas et al., 2020). Scientific claim verification similarly treats claims and evidence as structured objects that can be retrieved and checked (Wadden et al., 2020). These lines are relevant because ML artifacts often appear in comparative and evaluative statements. However, the thesis does not treat broad claim mining or leaderboard reconstruction as its main target. Instead, results and claims are used as context for the central thesis focus: concepts, artifacts, workflow relations, and evidence-linked records.

The choice of extraction targets constrains all later design decisions. It determines which schemas are useful, which corpora and benchmarks are comparable, which annotation guidelines are needed, which model outputs can be evaluated, and which records can support downstream exploration. For this reason, the next sections move from the target view to the resources, representations, and evaluation choices that make these targets operational.

2.4 Resources and Benchmarks for Scholarly IE

Scholarly IE resources define the empirical basis on which extraction tasks can be trained, evaluated, compared, and reused. They specify which documents are included, which parts of those documents are annotated, which targets are represented, which labels are available, and which outputs count as correct. Large scholarly corpora such as S2ORC provide machine-readable full text, metadata, references, citation mentions, and links to document objects, but they do not by themselves define an IE benchmark (Lo et al., 2020). Benchmarks add task-specific annotation layers, splits, evaluation assumptions, and documentation. For this thesis, resources and benchmarks are therefore not only data sources. They determine which kinds of scholarly objects and relations can be learned, which comparisons are meaningful, and which extraction results can later be used in evidence-linked exploration.

A first important resource dimension is document scope. Early scientific IE resources often focus on abstracts or selected text units, which reduces annotation cost and makes evaluation more controlled. ScienceIE defines a shared-task setting for extracting scientific keyphrases and relations from scientific publications (Augenstein et al., 2017). SciERC annotates scientific entities, relations, and coreference links in scientific abstracts and became a central benchmark for scientific IE (Luan et al., 2018). Such resources are valuable because they provide clear task definitions and compact evaluation settings. However, abstract-level and selected-sentence resources limit which evidence can be observed. Full-text and document-level resources address this limitation by including evidence from methods, experiments, appendices, captions, and other later parts of papers. SciREX formulates scientific IE as a document-level task over full papers with salient entities and n-ary relations (Jain et al., 2020). SciER continues this direction with full-text entity and relation annotations for datasets, methods, and tasks, including an out-of-distribution evaluation setting (Q. Zhang et al., 2024). This distinction is central

for the thesis because the later chapters require resources that support paper-level interpretation even when extraction components operate over local text units.

A second dimension is target coverage. Some resources annotate broad scientific entities, relations, and coreference links. Others focus on more specific scholarly objects, such as task–dataset–metric records, software mentions, or scientific claims and evidence. TDMSci provides a specialized corpus for task, dataset, and metric entity tagging in NLP papers (Hou et al., 2021). Related leaderboard-extraction work targets task, dataset, metric, and score information for constructing structured comparison views over scientific results (Hou et al., 2019). Software-focused resources treat software as a scholarly object that is often important for reuse but weakly represented in standard bibliographic metadata. Softcite provides a gold-standard dataset for software mentions in research publications (Du et al., 2021). Claim-oriented resources define a different benchmark family: SciFact represents scientific claims together with evidence-containing abstracts and rationales (Wadden et al., 2020). These resource families show that scholarly IE benchmarks differ not only in corpus size or domain, but also in the kind of scholarly knowledge they make visible.

A third dimension is schema granularity. Resources may cover similar scholarly evidence while encoding different representational views. One resource may use broad labels such as method, material, or task, while another may distinguish datasets, data sources, metrics, model-related entities, software, and generic mentions. For relation extraction, this issue becomes even sharper because relation labels encode assumptions about what counts as a meaningful link between two mentions. A relation between a method and a task, a model and a dataset, or a metric and a result may be represented as a separate relation type, as part of an n-ary record, or not represented at all. This means that two datasets can appear to address the same scientific IE problem while supporting different evaluation claims. This issue prepares the thesis discussion of cross-dataset comparison: resources are not neutral containers of labels, but designed views over scholarly evidence.

Resource construction also affects what a benchmark can support. Some datasets are manually annotated from scratch, which gives stronger control over labels and annotation criteria but limits scale. Other resources combine automatic pre-annotation, distant supervision, weak supervision, and manual correction. SciREX is an example of a resource that combines automatic and human annotation to support document-level scientific IE at larger scale (Jain et al., 2020). Weakly supervised IE uses structured resources or heuristics to produce noisy labels, which can increase coverage but changes what errors and generalization claims mean (Hoffmann et al., 2011). This construction difference matters because controlled resources support precise evaluation of fine-grained labels, while broader or weakly supervised resources can support larger-scale coverage with less direct control over label quality. The thesis takes up this distinction in the later chapters, where full-text resources are used to study fine-grained extraction, relation modeling, and cross-dataset transfer.

For relation extraction resources, negative cases and candidate construction are especially important. A benchmark is shaped not only by which positive relations are annotated, but also by which candidate pairs were considered and which non-relations were explicitly checked. If negative cases are sampled differently, or if only sentences containing target keywords are included, then precision, recall, and transfer results can change even with the same model architecture. Missing knowledge-base facts and

incomplete negative assumptions are known problems in distantly supervised IE (Ritter et al., 2013). More recent work on relation-extraction benchmark transparency also shows that data sources, instance selection, imbalance, documentation gaps, and leaderboard practices can strongly affect how benchmark results should be interpreted (Arzt et al., 2024). For this thesis, this point is relevant because relation extraction and cross-dataset comparison both depend on relation candidates, checked null cases, and annotation density.

Documentation is therefore part of the resource, not only a supplement to it. A benchmark should describe its motivation, corpus composition, collection process, preprocessing, annotation guidelines, label meanings, splits, limitations, and intended uses. Datasheets for datasets provide a general framework for documenting such properties in machine-learning datasets (Gebru et al., 2021). Data statements make a related argument for NLP by requiring clearer reporting of data provenance, population, collection conditions, and expected generalization limits (Bender and Friedman, 2018). For scholarly IE, such documentation is needed to understand what a label means, whether two datasets can be mapped, and whether an evaluation result reflects model quality or hidden resource differences. This becomes especially important for cross-dataset evaluation, where missing details about preprocessing, annotation scope, or candidate construction can make apparent performance differences hard to interpret.

In summary, scholarly IE resources are designed objects with consequences for learning, evaluation, transfer, and use. Document scope determines which evidence is visible. Target coverage determines which scholarly objects can be extracted. Schema granularity determines which distinctions are preserved or collapsed. Construction mode determines how much control the benchmark has over labels and negatives. Documentation determines whether the resource can be reused and compared. These resource choices prepare the later discussion of modeling approaches and the evaluation problem of comparing extraction systems across different scholarly IE datasets.

2.5 Modeling Approaches and LLM-Based Extraction

Modeling approaches connect the extraction targets and benchmark resources discussed above to predicted entities, relations, and structured records. For scholarly IE, the relevant modeling choices concern how systems decompose the task, represent spans, construct relation candidates, use supervision, and control the output schema. Some approaches separate extraction into modular steps, while others model entities and relations jointly or generate structured outputs directly. This section focuses on the modeling families most relevant for the thesis: modular pipelines, supervised joint and span-based extraction, scientific-domain encoders, LLM-based extraction, and hybrid schema-aware systems.

Pipeline architectures decompose IE into separate components, typically including entity detection, relation candidate generation, relation classification, filtering, normalization, and storage. This modular design remains useful because each component can be trained, evaluated, replaced, or inspected separately. It also fits practical scholarly IE systems, where extracted mentions may need later linking, evidence display, or manual review. Pipeline systems are sometimes treated as weaker than joint models

because entity errors can propagate into relation extraction. However, strong pipeline approaches can be competitive when entity and relation components use separate contextual encoders and when entity information is represented carefully for relation classification (Zhong and Chen, 2021). The main trade-off is therefore not simply pipeline versus end-to-end modeling, but the degree to which modularity, candidate construction, and error propagation are controlled.

Supervised joint and span-based models address this trade-off by representing entities and relations in a shared modeling framework. Instead of first committing to a final set of entity mentions and then predicting relations, these models often enumerate candidate spans, compute contextual span representations, and score entity types and relations over span pairs. DyGIE++ is a central example of this direction, using contextualized span representations for entity, relation, and event extraction across domains (Wadden et al., 2019). Such models are attractive for scholarly IE because entities and relations are closely connected: a span's type can constrain possible relations, and relation evidence can help disambiguate the role of a mention. At the same time, joint models depend strongly on annotation conventions. If span boundaries, nested mentions, relation signatures, or label granularity differ across resources, the same architecture can be trained and evaluated under different assumptions. This becomes especially relevant for fine-grained scholarly ERE, where model performance is tied to the schema and to the candidate spaces defined by the dataset.

Pretrained language models provide the contextual representations on which many supervised IE systems rely. For scholarly IE, domain adaptation is important because scientific texts contain specialized terminology, rare artifact names, long noun phrases, abbreviations, and reporting conventions that differ from general-domain text. SciBERT is a standard example of a pretrained encoder built on scientific publications and has been shown to improve performance across scientific NLP tasks (Beltagy et al., 2019). More generally, continued pretraining on domain and task data can improve downstream performance when the target domain differs from the original pretraining data (Gururangan et al., 2020). For the thesis, this supports the use of supervised encoder-based extraction as a strong basis for fine-grained scholarly NER and ERE when annotated data is available.

LLM-based extraction changes the implementation of IE by reformulating extraction as instruction following, question answering, or structured generation. Unified information extraction frameworks treat heterogeneous IE tasks as text-to-structure generation under a schema, allowing entities, relations, and events to be expressed in a common output format (Lu et al., 2022). Another line reformulates entity and relation extraction as question answering, where prompts or queries guide the model toward spans or decisions relevant to a target relation (X. Li et al., 2019). These formulations are useful for scholarly IE because task definitions can be expressed in natural language and adapted to new label sets with less task-specific architecture design. LLM-based scientific IE also shows that instruction-based extraction can produce structured representations from scientific text in specialized domains (Shamsabadi et al., 2024). Retrieval-supported prompting and example selection extend this idea by conditioning the model on similar examples, relation signatures, or relevant contexts, which is especially useful when labeled data is limited.

The flexibility of LLM-based extraction does not remove the need for controlled evaluation. Fine-grained scholarly IE often requires exact or near-exact spans, valid

labels, relation signatures, and evidence-grounded records. LLM outputs may be semantically plausible but difficult to align with benchmark annotations, especially when they paraphrase mentions, normalize names, omit boundary details, or produce unsupported records. Empirical work on LLMs for information extraction shows that prompting can be useful, but fine-tuned smaller models can still outperform LLM prompting in many strict IE settings, especially with fine-grained labels and increasing amounts of labeled data (Ma et al., 2023). Generative relation extraction also raises evaluation questions because generated relations must be assessed for factualness, granularity, completeness, and alignment with a target schema (P. Jiang et al., 2024). For scholarly IE, this means that LLM-based extraction is best treated as a complementary strategy rather than a replacement for supervised, schema-bound evaluation.

Hybrid and schema-aware systems combine these modeling families. A practical system may use a supervised model to detect candidates, an LLM to classify or verify difficult cases, retrieval to select relevant examples, and post-processing to normalize outputs. Such systems can benefit from the precision and efficiency of supervised components while retaining some flexibility of instruction-based extraction. LLM-based knowledge-graph construction work makes this pattern explicit by combining extraction, schema definition, schema retrieval, and post-hoc canonicalization (B. Zhang and Soh, 2024). For scholarly IE, schema-aware modeling is also important because resources may use different output spaces even when they cover related evidence. Models can therefore share encoders or intermediate representations while preserving dataset-specific heads, label spaces, or schema constraints.

Modeling choices cannot be judged independently from resources, schemas, annotation rules, and evaluation metrics. A pipeline may be preferable when modular inspection and downstream integration are central. A supervised joint model may be preferable when enough annotated data exists and strict span-level and relation-level evaluation are required. LLM-based extraction may be preferable when supervision is scarce, schemas change quickly, or task reformulation reduces the need for task-specific training. Hybrid approaches become useful when extraction needs both control and flexibility. The next section therefore turns to annotation, evaluation, and schema mismatch, where these modeling choices are linked to the conditions under which extraction quality can be measured and compared.

2.6 Annotation, Evaluation, and Schema Mismatch

In scholarly IE, evaluation is not only a scoring step after modeling. It is the point where target definitions, annotation rules, model outputs, and benchmark assumptions are made comparable. A model prediction can only be judged as correct relative to a schema, an annotation guideline, a candidate space, and a matching rule. This is especially important for fine-grained scholarly IE, where entities may have uncertain boundaries, relations may depend on several linked decisions, and datasets may encode related evidence through different label spaces. The later chapters take up these issues in entity extraction, relation extraction, and cross-dataset generalization settings.

Annotation guidelines define the effective task. They state which mentions should be annotated, which cases should be excluded, how labels are distinguished, and how annotators should handle borderline cases. Annotation platforms such as INCEpTION support this process through layered annotation, project management, machine assis-

tance, and adjudication workflows (Klie et al., 2018). However, annotation quality is not only a tooling issue. Inter-annotator agreement provides evidence about how consistently a schema can be applied, while disagreement often reveals ambiguous labels, underspecified rules, or difficult boundary cases (Artstein and Poesio, 2008). For scholarly IE, such ambiguity is common because papers contain technical noun phrases, generic artifact mentions, nested descriptions, and context-dependent roles.

Span-level evaluation makes these annotation choices directly visible. Exact span matching treats a prediction as correct only when the predicted boundary and label match the gold annotation. Relaxed or partial matching can count overlapping spans as correct, which may better reflect cases where the system identifies the right mention but not the exact boundary. Shared-task evaluation in biomedical IE has long used different matching conditions to separate boundary errors from type or relation errors (Segura-Bedmar et al., 2013). Nested and overlapping mentions add another difficulty because a single text region can contain several valid entities at different levels of detail (Alex et al., 2007). Relaxed matching is therefore useful for interpreting boundary uncertainty, but it can also hide errors that matter for relation extraction, linking, or downstream aggregation.

Relation evaluation is stricter because a relation prediction combines several decisions. A system must identify the relevant argument mentions, assign or inherit their entity labels, choose the correct relation label, and often preserve argument order or direction. SemEval relation classification tasks make this explicit by defining relations over ordered pairs of nominal arguments (Hendrickx et al., 2010). TACRED similarly represents relation instances through subject and object spans, entity types, relation labels, and no-relation cases (Y. Zhang et al., 2017). This means that a relation score can decrease because of a wrong span boundary, a wrong entity type, a wrong relation label, a reversed argument order, or an incorrect candidate pair. For fine-grained ERE, evaluation therefore depends on both entity-level and relation-level matching assumptions.

Negative examples and candidate construction are part of this evaluation problem. A missing relation is meaningful only if the candidate pair or sentence was actually considered during annotation or evaluation. If a benchmark contains many automatically generated negative pairs, relation extraction can become dominated by no-relation decisions. If negative cases are sparsely sampled, reported precision and recall can become difficult to compare with settings where all candidate pairs were checked. Re-TACRED shows that correcting relation labels and no-relation cases can change conclusions drawn from a widely used relation extraction benchmark (Stoica et al., 2021). This illustrates why relation benchmarks need careful reporting of candidate construction, class imbalance, annotation density, and null cases.

LLM-based and generative IE adds another evaluation layer. Unlike conventional span classifiers, generative models may produce normalized names, paraphrased spans, reformatted records, unsupported relations, or invalid structured outputs. These outputs must often be mapped back to benchmark units before they can be scored. For strict scholarly IE, this mapping is not a minor post-processing step because it determines whether the output satisfies the target schema and remains grounded in the source text. Work on generative relation extraction argues that exact reference matching alone can miss important dimensions of generated outputs, such as factualness, completeness, uniqueness, and granularity (P. Jiang et al., 2024). For this thesis, the important

consequence is that LLM-based extraction should be evaluated not only by whether it produces plausible records, but also by whether those records can be validated against spans, labels, relations, and evidence.

Schema mismatch is the central comparability problem across scholarly IE datasets. Resources may cover similar evidence while differing in label granularity, relation semantics, corpus scope, annotation density, and candidate construction. A label called *method*, *task*, or *dataset* may not mean the same thing across resources if the annotation guidelines define different inclusion rules. Likewise, relation labels that look similar can encode different assumptions about argument types, evidence locality, or event structure. Cross-dataset relation extraction work shows that generalization depends on label mapping and on whether relation categories are actually compatible across datasets (Arzt et al., 2025). Earlier work on learning across relation datasets also shows that shared modeling is difficult when corpora use different predefined relation schemas (Fu and Grishman, 2018).

Schema alignment can make controlled comparison possible, but it also changes the task. Mapping several resources into a shared label space can reveal which distinctions are stable across datasets and can support cross-dataset evaluation. At the same time, unification can remove distinctions that individual resources were designed to preserve. This is especially problematic for relation extraction, where collapsing entity types or relation labels may change the meaning of a relation instance. For this reason, schema alignment should be treated as an explicit methodological decision rather than a neutral preprocessing step. The relevant question is not only whether datasets can be mapped, but also what information is lost or preserved by the mapping.

Documentation is therefore necessary for interpreting evaluation results. Datasheets for datasets provide a general framework for documenting dataset motivation, composition, collection, preprocessing, intended uses, and limitations (Gebru et al., 2021). Data statements make a related argument for NLP by connecting data provenance and population information to the validity of empirical claims (Bender and Friedman, 2018). For scholarly IE, such documentation should include annotation guidelines, span rules, relation definitions, candidate construction, preprocessing, splits, and known limitations. These details determine whether extraction results can be reproduced, compared, transferred, and inspected in downstream applications. The next section therefore turns from evaluation conditions to applications and exploration systems, where extracted records must support evidence inspection, aggregation, and user-facing analysis.

2.7 Applications and Exploration Systems

Scholarly IE becomes useful when extracted entities, relations, and records can support downstream work with scientific literature. Applications need outputs that are structured enough to search and aggregate, but also traceable enough to inspect against the source text. This is especially important when extraction targets include research artifacts such as datasets, models, methods, software, tasks, metrics, and workflow relations. The preceding sections discussed how these targets are defined, annotated, modeled, and evaluated. This final section closes the related-work chapter by considering how extracted scholarly knowledge can be used in exploration systems.

One application area is discovery and aggregation across paper collections. Keyword

search can retrieve relevant documents, but it does not by itself organize papers by the artifacts, methods, tasks, or relations they contain. Extraction-based systems can turn these objects into facets, filters, and grouping units. Large scholarly knowledge graphs show this direction by representing research entities, claims, and relations from scientific publications in structured form. CS-KG, for example, constructs a large-scale computer-science knowledge graph from extracted research entities and claims and supports downstream services such as advanced literature search and trend analysis (Dessì et al., 2022). Graph-based exploration systems such as GrapAL similarly show how semi-automatically constructed scholarly knowledge graphs can support querying and exploration across scientific literature (Betts et al., 2019). For this thesis, the relevant point is not the graph representation itself, but the way extracted records enable paper-level and corpus-level exploration beyond document retrieval.

Relations are important because many exploration tasks require more than lists of mentioned artifacts. A system that only shows that a paper mentions a model and a dataset cannot tell whether the model was trained on the dataset, evaluated on it, compared against a baseline on it, or only discussed it as related work. Structured records and graph edges can make such differences explicit. They also make it possible to aggregate across papers, for example by grouping which datasets are used for a task, which methods are compared on a benchmark, or which software tools occur in experimental workflows. This also means that application quality depends on the same schema and evaluation choices discussed above. If relation types are too coarse, applications lose useful distinctions; if they are too fine-grained or inconsistently extracted, aggregation becomes harder.

Reproducibility-oriented applications provide a second motivation for artifact-centered IE. Scientific work often depends on datasets, software, models, metrics, preprocessing choices, and evaluation settings that may be described only in the paper text. Extracting these objects and their relations can help users inspect what was used, how it was connected, and where reported results depend on particular artifacts. Software-focused knowledge graphs illustrate this application direction. A large-scale analysis of software mentions in PubMed Central shows how extracted software mentions and metadata can support analyses of software use, citation practices, provenance, and visibility in scientific publications (Schindler et al., 2022). For ML-centered scholarly IE, the same logic extends to datasets, model variants, benchmarks, and experimental links.

Evidence inspection is necessary because extracted records are imperfect and often context-dependent. Applications should therefore not only show an extracted entity or relation, but also the sentence, passage, or paper section that supports it. Scientific claim verification makes this requirement explicit by pairing claims with evidence-containing abstracts and rationale sentences (Wadden et al., 2020). Although claim verification is an adjacent task, it illustrates a general requirement for scholarly IE applications: users need to move from an aggregated record back to the textual evidence that justifies it. This is especially important for relation extraction, where a predicted link may depend on local wording, section context, or the interpretation of an experimental setup.

Interactive review and human validation can further improve the usefulness of extraction-based systems. When automatic extraction is uncertain, users may need to confirm, reject, or correct extracted statements. TinyGenius provides a relevant example by validating NLP-extracted scholarly knowledge statements through microtask

crowdsourcing and storing validation decisions as provenance (Oelen et al., 2024). This kind of workflow shows why user-facing systems need representations that expose enough context for review. For this thesis, interactive review is not the main methodological focus, but it motivates the requirement that extracted artifacts and relations remain inspectable rather than being reduced to opaque aggregate counts.

The application scope of this thesis is therefore narrower than general scholarly search, recommender systems, citation recommendation, or full systematic-review automation. Its focus is extraction-based exploration: using structured, evidence-linked records to help users inspect and compare machine-learning artifacts and workflows across papers. This closes the chain developed in this chapter. Extraction targets define what should be represented; resources and annotations determine what can be learned and evaluated; models produce candidate records; schema and matching choices determine comparability; and applications require those records to remain searchable, aggregable, and connected to textual evidence.

3

Fine-Grained Entity Recognition for Machine Learning Research

Publication Information

Title: GSAP-NER: A Novel Task, Corpus, and Baseline for Scholarly Entity Extraction Focused on Machine Learning Models and Datasets

Authors: Wolfgang Otto, Matthäus Zloch, Lu Gan, Saurav Karmakar, and Stefan Dietze

Status: Published in *Findings of EMNLP 2023*

Reuse: This chapter reproduces the text of the publication listed above in thesis layout.

Context

This publication is the first methodological building block of the thesis and sets up the basis for the later chapters. The overall goal of the thesis is to enable automated extraction of structured, machine-learning specific information from scholarly publications, so that ML-based research becomes easier to compare, check, and reuse. GSAP-NER targets this goal at the entity level by defining machine learning models, model architectures, datasets, and related concepts as explicit entity types that can be detected in text.

By introducing a dedicated NER task and a manually annotated full-text corpus, this work addresses a main limitation of earlier scholarly IE resources: many existing datasets are too coarse-grained and too abstract-focused to capture how ML artifacts are actually written about in papers. The focus on full-text annotation and on informal mentions matches the thesis-wide aim of working with realistic paper writing and supporting downstream use.

The GSAP-NER corpus and baseline models are reused and extended in later contributions. In particular, the entity schema and annotation principles are carried over to GSAP-ERE, where entity recognition is combined with relation extraction. In addition, the lessons from GSAP-NER on ambiguity, context effects, and reporting patterns inform the later chapters on LLM-based extraction and on interactive exploration of extracted ML knowledge. Overall, this chapter anchors the thesis in a shared and tested representation of ML-related entities in scholarly text.

Abstract

Named Entity Recognition (NER) models play a crucial role in various NLP tasks, including information extraction (IE) and text understanding. In academic writing, references to machine learning models and datasets are fundamental components of various computer science publications and necessitate accurate models for identification. Despite the advancements in NER, existing ground truth datasets do not treat fine-grained types like *ML model* and *model architecture* as separate entity types, and consequently, baseline models cannot recognize them as such. In this paper, we release a corpus of 100 manually annotated full-text scientific publications and a first baseline model for 10 entity types centered around ML models and datasets. In order to provide a nuanced understanding of how ML models and datasets are mentioned and utilized, our dataset also contains annotations for informal mentions like "our BERT-based model" or "an image CNN". You can find the ground truth dataset and code to replicate model training at `https://data.gesis.org/gsap/gsap-ner`.

3.1 Introduction

Throughout various disciplines, the scientific process constantly produces new knowledge, innovative discoveries, and valuable insights, which typically are published in conference proceedings and journal articles. The increasing volume of scholarly artifacts underscores the importance for scientists to efficiently locate, comprehend, and utilize these resources in their daily work. Consequently, the NLP community is constantly creating methods to extract named entities from the scholarly domain, recognizing its significance in facilitating scientific understanding.

With the advent of artificial intelligence (AI), the landscape of machine learning (ML) approaches has evolved, incorporating techniques such as deep learning (DL) and fine-tuning of large language models (LLM). Consequently, to effectively comprehend scientific articles related to ML, AI, or data science, it becomes crucial to identify and comprehend these emerging entity types.

Developing effective NER models for these entities requires annotation guidelines and ground truth datasets to train robust language models (QasemiZadeh and Schumann, 2016; Luan et al., 2018). However, existing guidelines and ground truth datasets for

MLModelGeneric: Our model dubbed MLModel: XLM-R, significantly outperforms MLModel: multilingual BERT (MLModel: mBERT) on a variety of DatasetGeneric: cross-lingual benchmarks, including +14.6% average accuracy on Dataset: XNLI, +13% average F1 score on Dataset: MLQA, and +2.4% F1 score on Dataset: NER.

Figure 3.1: Example of machine learning models and dataset related mentions annotated according to our tag set. "Our model" and "cross-lingual benchmarks" are considered *informal mentions*, whereas the others are noun phrases mentioning named entities.

scholarly entities have not adequately addressed the finer-grained entity types, such as *ML Models* and *Datasets* as distinct entities. Instead, state-of-the-art works treat ML models as *Methods* (Färber et al., 2021), failing to differentiate between the model instance, type, and underlying architecture. Similarly, dataset mentions are typically categorized as *Material*, overlooking the fact that this can also encompass knowledge bases, resources, or other corpora.

Based on our ground truth we fine-tune a first baseline model for our task of ML model and dataset named entity recognition. We employ three state-of-the-art baseline models for that, SciBERT (Beltagy et al., 2019), RoBERTa (Y. Liu et al., 2019), and a more recent pre-trained language model called SciDeBERTa-CS (Jeong and Kim, 2022). We have found that SciDeBERTa-CS performs best on the entity types MLModel and Dataset, with an F1 score of 0.71 and 0.81, respectively.

Creating a sizable ground truth dataset like GSAP-NER is costly in terms of effort. Therefore, as a final experiment, we explore the minimum quantity of fully-annotated texts required to observe a noteworthy improvement in performance by incrementally increasing the size of the training data. Our dataset enables researchers and practitioners to extract precise and domain-specific information, contributing to fields such as information retrieval, scientific knowledge mining, automated literature analysis, and knowledge graph creation.

Our research presents four key contributions, each aimed at advancing the state of scholarly entity and concept detection:

- We provide a manually annotated dataset containing 100 full-text computer science publications with over 54k entity mentions in 25,857 sentences (Section 3.3).
- We introduce a fine-grained tag set designed for detecting scholarly entities and concepts, customized to reflect the use and presentation of machine learning models and datasets in scientific publications (Section 3.3.1).
- We conduct a comprehensive performance evaluation of baseline models for our ten defined entity types (Section 3.4 and 3.5).
- We explore the minimum number of annotated publications needed to achieve satisfactory performance in our fine-grained scholarly NER task, which can guide future annotation projects (Section 3.6.2).

All materials, such as the ground truth dataset and the code to replicate model training can be found at `https://data.gesis.org/gsap/gsap-ner`.

3.2 Related Work

Among the works dealing with the task of scholarly information extraction, in this section we focus on those named entity recognition[1] methods which are machine-learning-based and not rule-based approaches. As a general overview, Nasar et al. (2018) gives a comprehensive list of approaches on information extraction from scientific publications.

Multiple datasets serve as ground truth datasets for Named Entity Recognition (NER), each catering to specific tasks. Please consult Table 3.1 for a comparison of the most related ground truth datasets to ours, GSAP-NER.

[1] In the literature, it is also referred to as key-insight extraction, (typed) entity recognition, entity extraction, or (scientific) concept extraction.

	SciERC	SciREX	TDMSci	Heddes	GSAP NER
Ann.unit	♣	♥	♠	♠	♥
# pub.	500	438	n/a	n/a	100
# pos. sent.	2,558	76,223	2,000	2,911	25,857
# neg. sent.	214	44,209	0	3,089	7,530
# ent.types	6	4	3	1	10
# mentions	8,089	156,931	2,937	3,729	54,598
# ment./pub.	16.2	358.3	n/a	n/a	546.0
Material	■	□	□	□	■
Dataset	□	■	■	■	■
DataSource	□	□	□	□	■
Metric	■	■	■	□	□
Method	■	■	□	□	■
ML Model	□	□	□	□	■
ModelArch.	□	□	□	□	■
Task	■	■	■	□	■

Table 3.1: Comparison of ground truth datasets for scholarly NER Tasks including annotated entity types.
Annotation units: ♣=abstract, ♠=sentence, ♥=full-text.

Among datasets working on abstracts, SciERC stands out as a prominent dataset, comprising 500 abstracts extracted from 12 AI conference and workshop proceedings (Luan et al., 2018). This rich dataset contains annotations for scientific entities, their relationships, and co-reference clusters, which are invaluable for related NLP tasks.

Another dataset, SciREX, offers comprehensive coverage with 438 fully annotated documents, specifically targeting mention identification and relationship extraction of entities related to methods, tasks, datasets, and metrics (Jain et al., 2020). To prepare their ground truth dataset for the NER task, they combined distant supervision and manual correction of automatically pre-annotated full-texts. In contrast to their work, we created a fully manually annotated corpus. This enables us to define our tag set independently of current approaches and to avoid potential model bias introduced by pre-annotation.

Hou et al. (2021) contributed significantly to this area by presenting TDMSci, a corpus containing domain expert annotations for TDM entities in 2000 sentences extracted from NLP papers, alongside a dedicated TDM tagger designed for this specific task.

Heddes et al. (2021) developed a ground truth dataset for dataset mention detection, comprising 6,000 annotated sentences selected by the occurrence of dataset related word patterns that were sourced from four major AI conference publications. Approximately half of them containing one or more named datasets.

An emerging NLP task known as leaderboard extraction focuses on extracting Task-Dataset-Metric-Score (TDMS) tuples from scholarly papers, enabling the generation of an aggregated comparison view of the main entities of interest (Kabongo et al., 2021).

Annotation sentence with identified spans	**Justification**
For the ResNets we train a ResNet-50, a ResNet-101, and then 3 more ...	the ResNets is annotated as a whole as informal mention of multiple concrete *MLModel*s. Additionally, it includes the nested annotation ResNets, which corresponds to the structural type information *ModelArchitecture*. ResNet-50 and ResNet-101 are actual executable *MLModel*s.
We publicly release a new large-scale dataset, called SearchQA... (an existing question-answer pair is) crawled from J!Archive, and augment it with the text snippets retrieved by Google.	*Dataset* and *DatasetGeneric* are identified with similar reason to the previous example, while unstable data source information are recognized as *DataSource*.

Table 3.2: Annotation examples for machine learning model related entities: MLModel, MLModelGeneric, ModelArchitecture. And data related entities: Dataset, DatasetGeneric, and DataSource.

Along this direction, Kardas et al. (2020) introduced an extraction pipeline, AxCell, for extracting results from tables listed in scientific articles. In 2022, D'Souza and Auer (2022) created CS-NER, a corpus of contribution-centric information extraction targets, namely research problem, method, solution, dataset, metric, and more.

Recent lines of research have explored end-to-end frameworks based on NLP extraction tasks, such as NER, which involve a series of interconnected methods aimed at creating knowledge bases or knowledge graphs. Agrawal et al. (2019) focused on extracting the aim, method, and result sections from scientific articles, utilizing this information to construct a scientific knowledge graph. Similarly, Mondal et al. (2021) developed *SciNLP-KG*, a framework designed to extract TDM entities and relations from papers in the NLP domain. Furthermore, Dessì et al. (2022) presented a computer science knowledge graph (CS-KG) that is automatically generated and periodically updated. They achieved this by applying an information extraction pipeline to a vast repository of research papers, offering a comprehensive and up-to-date resource for the computer science domain.

3.3 Dataset

Machine learning models and datasets are essential scholarly entities that are discussed in various scientific disciplines. For named entities, it is frequently observed that, depending on context, identical string spans refer to different semantics. Take "BERT" in natural language processing as an example: it can refer to a particular pre-trained model with fixed parameters like "BERT Base" or its architecture, depending on the context. In addition to named entities, unnamed or informal mentions of machine learning models or datasets are more common in scientific text. But those informal mentions often contain nested references to other named entities and thereby carry extra information linking not only to machine learning models or datasets but also to

The average token count per paragraph stands at 109.1, with a median of 88 tokens. Notably, only 16 paragraphs exceed the threshold of 512 tokens, as determined by tokenization using the SciBERT tokenizer. This enables their usability across various language models.

We transform the annotated spans into token wise labels based on the BIO tag scheme. As described in Section 3.3, our annotation guideline allows nested annotations, which presents challenges for out-of-the-box NER models. This is because the task of classifying each token becomes a multi-label classification problem rather than a multi-class classification problem. Analyzing on our GSAP-NER corpus showed two major patterns of nested annotations. The first involves named entities nested in *Generic*s, while the second relates to benchmark entities that are double annotated for both *Dataset* and *Task*. To address the first case, we split the entity tag set into two parts: *Generic* mentions and all other type of mentions. Additionally, we simplify the double annotated benchmark entities by converting them into a single *Dataset* span in the training set. This approach enables us to use two separate models for each of the tag set, resolving the nested entity annotation problem An analysis showed that the simplifications need to be done for generating the training data leads to a upper bound of 98.7% F1 score when testing the performance of the simplified tag set on the full annotations without any simplification. While fine-tuning PLMs we use a final fully connected layer on top of the encoding layer and trained to predict one label for each token using a cross entropy loss for each of the two models.

3.5.2 10-fold Cross Validation

The broader perspective of our model is to solve the NER task on whole documents, even if the unit or processing for our model is one paragraph. We consider 10-fold cross validation where folds are created such that all paragraphs from one publication are present in the same fold. In each cross validation round 80% of publications are used for training, 10% for validation and the remaining 10% for testing. For reasons of reproducibility, we publish the publications used in each fold as part of the data set.

3.5.3 Metrics

We evaluate our models with entity-level F1 score, where each entity annotation is identified by the *paragraph id*, *start index*, *end index*, and *label*. Gold annotations and predictions are represented as sets of entity annotations, and the F1 score is calculated based on these sets. For comparison, we also employ partial-match F1 score, which considers a predicted entity span as a match if it overlaps with a gold annotation of the same label. This metric is particularly useful for assessing the correct position of entity annotations, even if the beginning and end of the tag span are not precisely predicted. To measure performance for each entity type separately, we only consider gold annotations and predictions for a specific label.

3.6 Experimental Results

3.6.1 Baseline Models

Fine-tuned PLMs proved to perform well on NER tasks for scholarly document processing. The F1 score performance overview of the four compared models (i.e., SciBERT, SciDeBERTa-CS, RoBERTa-Base, and RoBERTa-Large) show a general applicability on the given task. With exact-match F1 score in the range from 61.9 to 64.4 (Table 3.5), the performance is comparable with other annotation approaches (Luan et al., 2018). The best performing model SciDeBERTa-CS outperforms the much bigger RoBERTa model for nearly every entity type. Nonetheless, the performance varies in terms of entity types and models. To assess the significance of SciDeBERTa-CS model's performance enhancements relative to the other models in our ten-fold cross-validation setup, we conducted a paired t-test. The obtained p-values for all comparisons were found to be below the significance threshold (commonly set at 0.05), indicating a statistically significant difference in performance.
To characterize the similar performance differences among all models for different labels, we identified two distinguishing criteria for entity types. Firstly, concrete named entities (e.g., MLModel: 70.1% F1 and Dataset: 81.7% F1) exhibit superior performance in contrast to conceptual entities (e.g., Method: 47.6% F1 or ModelArchiteture: 33.9% F1), irrespective of the quantity of training samples. Notably, the Method entity type is the most prevalent, encompassing over 12,500 annotated text spans (as shown in Table 3.4). The second criteria employs the presence of structural anchor points to distinguish between entity types. For instance, standardized patterns such as citations can be easily identified with a supervised approach. The existing weak URL extraction performance can be attributed to the limited number of annotations and the fragmented nature of URLs generated during the conversion from PDF to text. Inaccuracies in rare URL annotations can significantly increase the error rate, whereas fragmented URLs pose challenges in accurately detecting the beginning and end of a URL. Another valuable insight arises from the observed variation in performance between exact-match and partial-match metrics. The performance gap ranges from +2.9% on MLModel to +14.4% on Method, emphasizing the significance of distinguishing between concrete and conceptual entity types. Recognizing the correct text spans for conceptual entity types remains a challenge for the models due to the uncertainties among the annotators. Please refer to Table 3.3 for comparison.

3.6.2 Train Size Experiment

In comprehensive full-text annotation initiatives, the quantity of annotated full-texts assumes a pivotal role owing to the substantial cost advantages linked with annotating a smaller subset of publications. To explore this facet, we executed an experiment aimed at assessing performance metrics across various training set dimensions. For the sake of efficiency, the SciBERT model was used in this experiment. In our 10-fold setup, we conducted fine-tuning for ten distinct models within each fold. In every iteration, a designated proportion of documents from the training dataset was allocated to individual models. The number of training documents varied in each iteration, ranging from 8 to 80, with increments of eight. The findings from this experiment are visually represented in Figure 3.2.

When assessing the F1 score, we observed a substantial standard deviation in the case of models with a limited number of publications within the training dataset. Conversely, for models trained with more than 24 documents, resulting in a dataset of greater variability, this phenomenon was markedly mitigated. Subsequently, our investigation revealed that the fluctuations in the exact-match F1 score demonstrated a diminishing trend, stabilizing after the incorporation of more than 40 publications for training.

3.7 Conclusion

We introduce GSAP-NER, a manually annotated corpus over full-text scholarly publications from the computer science domain, designed for information extraction of ML models and datasets. By distinguishing ML models from methods and datasets from materials, our dataset enables researchers and services to gain deeper insights into the specific methods and materials employed in computer science research. We utilized our data and fine-tuned three state-of-the-art baseline models. The experiments showed that SciDeBERTa-CS reaches best performance on the majority of entities types, with an overall F1 score of 0.64 and 0.73 on exact span matches and partial span matches, respectively.

Despite the challenges involved in its creation, we believe GSAP-NER remains a valuable resource for the development, evaluation, and benchmarking of NER models in the computer science domain. It offers researchers and practitioners a comprehensive and domain-specific dataset, addressing the limitations of existing datasets that often

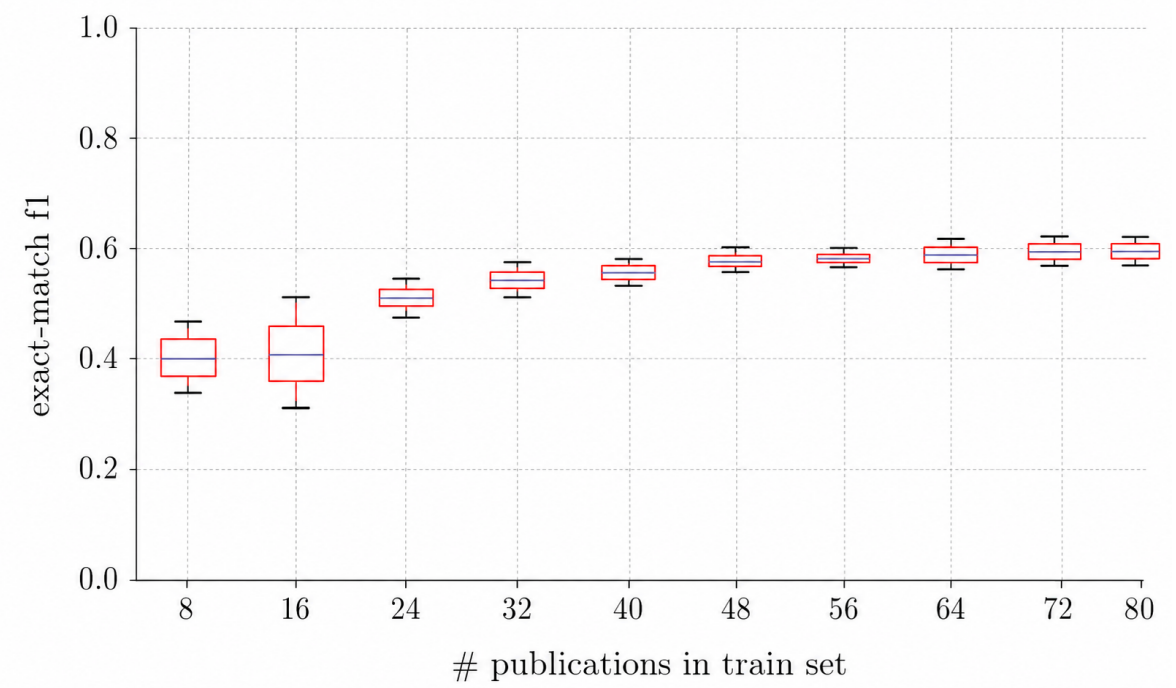


Figure 3.2: Increasing overall exact-match F1 performance of the SciBERT model trained on varying number of publications. The train set size in the 10-fold set up ranges from 8-80 publications in every fold. The boxplot illustrates the performance differences across folds.

lack specialized entity differentiation. Furthermore, this dataset can contribute to advancing research in areas such as information retrieval, scientific knowledge mining, automated literature analysis, and knowledge graph creation.

As future work, we aim to employ a multi-stage model training approach and leverage additional background knowledge to disambiguate syntactic mentions from their semantic context. Such background knowledge could come from ML ontologies and knowledge graphs, such as the ORKG or CS-KG. We also envision exploring entity relationships, co-reference resolution, and entity attributes as future directions to enhance the value of this dataset.

Limitations

Despite our diligent efforts, developing a gold standard dataset for entity extraction using a fine-grained and comprehensive tag set focused on machine learning models and datasets remains a non-trivial undertaking. This leads to the following limitations associated with the creation of our corpus. First, our work suffers from low interrater agreement on certain entity types, and thus, the model performs poor on those types. For instance, when frequently used models such as "RoBERTa" are mentioned, it can be difficult to determine whether to classify them as ModelArchitecture or MLModel, depending strongly on the context. Efforts to address ambiguous types in the annotation guideline or increased training time of annotators did not solve this issue. Second, the paper selection is conducted within the machine learning domain and does not include infrequent publication types, such as surveys or reproducibility studies. Furthermore, the potential applicability of our approach across various research domains remains a topic for future investigation. Finally, during the model training process, we excluded paragraphs that were identified as erroneous by the annotators. It is essential to address and resolve the resulting issues before the model can be effectively used in a productive real-world setting.

Ethics Statement

The authors foresee no ethical concerns with the work presented in this paper.

Acknowledgements

We thank the anonymous reviewers for their constructive feedback and intense rebuttal phase. This work has been partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of the Projects BERD@NFDI (grant number 460037581), NFDI4DS (grant number 460234259), as well as Unknown Data (grant number 460676019).

4

LLM-Assisted Extraction of Software Information

Publication Information

Title: Enhancing Software-Related Information Extraction via Single-Choice Question Answering with Large Language Models

Authors: Wolfgang Otto, Sharmila Upadhyaya, and Stefan Dietze

Status: Published NSLP 2024 workshop paper (Archival, Peer-Reviewed)

Reuse: This chapter reproduces the text of the publication listed above in thesis layout. The original publication uses the term generative language model (GLM), while the thesis generally uses large language model (LLM). In this chapter, both terms refer to the same prompting-based extraction setting.

Context

This publication expands the thesis from supervised, schema-driven extraction towards extraction methods that rely on large language models with in-context learning. It uses the Shared Task on Software Mentions Disambiguation (SOMD) as a realistic setting where labeled data is limited and where software mentions and their attributes must be extracted reliably from scholarly text. The chapter therefore adds a second methodological angle to the thesis: how far LLM-based pipelines can support scholarly information extraction when task formulations and prompts are designed carefully.

Methodologically, the work reframes relation extraction as single-choice question answering and combines this with retrieval-based selection of relevant context examples. This connects to the thesis goal in two ways. First, it targets an important artifact type for reproducibility and reuse (software) that is often mentioned in inconsistent ways in papers. Second, it shows how task reformulation and retrieval can make extraction more stable in low-data conditions, complementing the supervised baselines used in the other chapters.

Within the thesis flow, this chapter provides evidence about strengths and limits of LLM-based extraction when compared to fine-tuned models, and it motivates why robust datasets and evaluation setups remain necessary. It also informs later parts of the thesis that focus on connecting results across datasets and on building systems that make extracted information accessible to users.

Abstract

This paper describes our participation in the Shared Task on Software Mentions Disambiguation (SOMD), with a focus on improving relation extraction in scholarly texts through Generative Language Models (GLMs) using single-choice question-answering. The methodology prioritises the use of in-context learning capabilities of GLMs to extract software-related entities and their descriptive attributes, such as distributive information. Our approach uses Retrieval-Augmented Generation (RAG) techniques and GLMs for Named Entity Recognition (NER) and Attributive NER to identify relationships between extracted software entities, providing a structured solution for analysing software citations in academic literature. The paper provides a detailed description of our approach, demonstrating how using GLMs in a single-choice QA paradigm can greatly enhance IE methodologies. Our participation in the SOMD shared task highlights the importance of precise software citation practices and showcases our system's ability to overcome the challenges of disambiguating and extracting relationships between software mentions. This sets the groundwork for future research and development in this field.

4.1 Introduction

The evolution of information extraction (IE) in scholarly communications requires innovative methodologies to accurately identify and categorise software mentions. This is a critical component for ensuring research transparency and reproducibility. The Shared Task on Software Mentions Disambiguation (SOMD) highlights the importance of refined citation practices amidst inconsistent referencing of software artifacts. This research leverages Generative Language Models (GLMs) to address the challenges in software mention extraction and relation identification, marking a significant step towards the sophisticated extraction of software-related information. Our approach integrates Retrieval-Augmented Generation (RAG) techniques with GLMs to dissect and comprehend the intricate web of software citations and their attributive details (e.g. the developer or the version) within scientific texts. By examining the potential of GLMs in performing IE tasks by transforming them into single-choice question-answering, we present a comprehensive analysis that addresses the nuances of software mention extraction and could be applied on a broader scope of scholarly artifacts NER and relation extraction. This paper explores the complexities of applying GLMs to NER tasks, providing insights into the challenges and proposing a new methodology for relation extraction that could pave the way for future innovations in the field.

4.2 Related Work

This section explores works in NER and IE, focusing on software mention extraction, scholarly artifact NER, and the use of GLMs in these areas.

In software mention extraction, the Softcite dataset (Du et al., 2021) and the SoMeSci knowledge graph (Schindler et al., 2021) are significant. Softcite offers a gold-standard dataset for extracting software mentions from biomedical and economic research, while SoMeSci Knowledge Graph includes software mentions in scientific

articles with relation labels such as version, developer and citations, highlighting the need for accurate software mention extraction.

In scholarly artifact NER, Saji and Matsubara, 2022 introduced a method using academic knowledge graphs to extract research resource metadata from scholarly papers, enhancing metadata quality and repository size. Otto et al., 2023 developed the GSAP-NER corpus for extracting machine learning-related entities from scientific publications, filling the gap in general-purpose NER models. These words demostrate the importance of domain-specific NER tasks and leverageing knowledge graphs for enhancing research resource repositories.

Regarding GLMs for NER, S. Wang et al., 2025 explores text-generation models for sequence labeling and underlying challenges, particularly in low-resource and few-shot setups. Furthermore, Xie et al., 2024 investigates LLMs' self-improving capabilities for zero-shot NER.

For GLMs in relation extraction, Wadhwa et al., 2023 uses few-shot prompting and fine-tuning with large language models achieving state-of-the-art performances in Relation Extraction using LLMs, while Wan et al., 2023 introduce GPT-RE enhancing relation extraction accuracy through task-specific entity representations.

Lastly, Xu et al., 2024 conducts a comprehensive survey providing an overview of generative information extraction using Large Language Models, categorizing works by IE subtasks and learning paradigms, and highlighting the transformative potential of GLMs in IE.

Overall, these studies underscore the evolving methods and significant contributions in extracting information from scholarly texts using specialized datasets, domain-specific approaches, and advanced generative models.

4.3 SOMD Shared Task

The Shared Task on Software Mentions Disambiguation (SOMD) aims to enhance transparency and reproducibility in scientific research by improving software citation practices. Participants are required to develop models that can identify and disambiguate software mentions in scholarly texts using the expanded Software Mentions in Science (SoMeSci) knowledge graph, with a focus on AI and Computer Science.

Subtask 1, Software NER, involves identifying four types of software-related entities: Application, Plugin, Operating System, and Programming Environment. Accurately classifying software mentions is necessary to understand their role in academic discourse.

Subtask 2, Attribute NER, aims to extract ten different attributive information associated with software entities, such as alternative names, abbreviations, authorship (developers), software release maintenance (versions, extensions, release dates, licenses), and elements of the referencing system (in-text citations, URLs, and software coreferences for cross-sentence linkage), to a specific software mention.

Subtask 3, Relation Extraction, involves establishing relationships between software entities and their attributes using specific relation types and mapping each attribute to these relations. This task aids in comprehending the interrelations of software entities within scholarly texts.

The tasks **evaluation metric**, is the Weighted Average Macro F1 score. This metric adjusts the influence of each label on the final result based on the number of test

instances assigned to it, ensuring that labels with fewer instances have a proportionally smaller impact on the overall performance evaluation.

4.4 Using GLMs for Software-Related IE Tasks

Retrieval-Augmented Generation (RAG) significantly bolsters the capabilities of Large Language Models (LLMs) in Information Extraction (IE) tasks by effectively utilizing both unstructured and structured contexts (Gao et al., 2024; Lewis et al., 2020). This dual approach is essential in IE for achieving precise extractions, yet the selection of optimal samples for the generative process poses a substantial challenge. Addressing this involves two primary strategies: utilizing sentence embeddings to find contextually similar textual content for the LLM, and identifying analogous entities to uncover beneficial training sentences. These methods enable the LLM to discern structural and semantic patterns for more accurate text extractions. A crucial obstacle is accurately identifying target entities within test instances, for which we employ a pre-trained Language Model (LM) tailored to our NER task. This LM is instrumental in both spotting potential entity candidates and facilitating entity similarity searches, leveraging last hidden state embeddings from training examples to locate matching entities within the dataset.

Our methodology extends to evaluating various retrieval techniques and their impact on the GLM's efficiency, particularly within a Few-Shot learning framework. We explore different methods, including random illustrative samples, text similarity-based RAG, and entity-based sentence retrieval, to provide the GLM with contextually relevant examples, thereby optimizing the software entity extraction process from scientific texts. This exploration aims to identify the most effective strategies for utilizing GLMs in domain-specific software entity extraction tasks. By overcoming challenges related to resource demand, execution time, and precision in entity identification, our approach aims to enhance the accuracy and efficiency of NER processes, contributing valuable insights and methodologies to the field of computational linguistics and information extraction.

A special retrieval method is used for the RE Task. Because for this task, the entities for each test instance are given, we easily could list all possible relations (compare Table 4.8.1). We could use them to find similar relations in the train set. If we find more then one, we decide for the one, with highest sentence similarity.

For the RE Task, a special retrieval method is employed. As entities for each test instance are provided, all possible relations can be listed based on Table 4.8.1. These relations can then be used to find relations of the same type, and with the same domain and range in the train set. If multiple similar relations are found, the one with the highest sentence similarity is selected.

4.4.1 Extraction of Software Entities

The extraction of software entities from scientific texts represents a specialized challenge within the realm of Named Entity Recognition (NER), targeting the identification of software entities across four distinct categories ranging from applications to operating systems. Furthermore, this task extends beyond mere identification, seeking to understand the intent behind each mention of software entities—whether it pertains to

creation, usage, deposition, or mere citation.

We address this challenging task using Generative Language Models (GLMs), despite their high demands on resources and time, especially when processing extensive publications with sparse relevant text. Our approach includes a pipeline strategy that prioritizes selecting relevant text passages for GLM analysis, improving efficiency by filtering out unrelated content. This method's success depends on the selection accuracy, directly impacting recall. However, the trade-off for reduced computational costs justifies the potential minor decrease in recall. Our performance optimization employs a hybrid method, combining a fine-tuned NER model for sentence selection with GLMs for information extraction. This approach faces limits, notably when sentences crucial for analysis are missed in the selection phase, capping the GLM extraction phase's accuracy as indicated by an initial sentence classification task recall of 0.882 (0.884 F1). This establishes a theoretical limit on extraction accuracy due to potential false negatives, illustrating a balance between efficiency and the precision constraints of GLMs in detailed text analysis.

4.4.2 Extraction of Software Attributes

Following the identification of software entities within scientific texts, a further nuanced aspect of Named Entity Recognition (NER) and Information Extraction (IE) tasks emerges in the extraction of associated software attributes. These attributes encompass a wide array of specific details, the version, developer, citations, URLs, release dates, abbreviations, licenses, extensions, software co-references, and alternative names. We used a similar approach as for subtask 1, and utilised train sample retrieval to augment the task description in a few shot setup. For each sample, including those derived from few-shot learning, the process entails presenting the sentence containing the software entity(ies) and then predicting a JSON list of identified entities along with their respective attribute types.

4.4.3 Relation Extraction as Single-Choice Question Answering Task

In the field of Natural Language Processing (NLP), the extraction of relations between entities within a text corpus poses significant challenges. This study proposes a novel approach by conceptualizing the task of relations extraction as a single-choice question-answering (QA) activity. This method entails generating a comprehensive list of all possible entities within a sentence, drawing from the existing entities and their relationships as delineated in the training dataset. Each potential pair of entities is then evaluated to ascertain if a specific relation, such as "version_of", appropriately links them. For instance, considering the relationship "version_of", a sentence may be formulated as "8 is the version of SPSS", representing a possible relation between the version number and the software entity.

For every sentence in the dataset, this process yields a set of single-choice questions, each positing a potential relationship between entities. These questions are then presented to a Large Language Model (LLM) for answering. The LLM's task is to select the most plausible relation from among the given options, thereby facilitating the extraction of accurate entity relations from the text. However, this approach is not

without its challenges. A primary source of error stems from instances where multiple relations could plausibly link a pair of entities, leading to ambiguity and complicating the single-choice question-answering framework. Despite the challenges, we demonstrate that treating relation extraction as a single-choice QA task provides a structured and innovative approach to extracting valuable insights from complex textual data.

4.5 Experiments

4.5.1 Models

Fine-tuned Model In our experimental setup, the initial phase focused on fine-tuning a language model specifically for Subtask 1, which involved Named Entity Recognition (NER). Following the methodology similar to Schindler et al., 2021, we employed the SciBERT model (Beltagy et al., 2019), given its pre-training on scientific corpus, making it apt for NER fine-tuning within scholarly texts. The fine-tuning process entailed rigorous parameter optimization, including adjustments to the batch size, learning rate, and the relative share of negative samples—sentences that do not contain any annotations. This optimization utilized a 90/10 train/evaluation data split from the available training dataset. Subsequent to parameter tuning, we conducted a final training run with a modified data split of 95/5 train/evaluation to maximize the training data's utility.

Generative Large Language Models For a comparative analysis, our study integrated Generative Large Language Models (GLLMs), specifically examining the performance differences between GPT-3.5-fast and GPT-4-fast models accessed via the OpenAI API. To ensure deterministic outputs for comparison, we set the temperature parameter to zero, eliminating randomness in the model's response generation.

4.5.2 Prompting

Software NER The prompting strategy for Software NER involved providing a concise task description, which included specifying the task as NER and intention classification, alongside an enumeration of target labels. We also highlighted the domain specificity of the texts (i.e., scientific publications) and requested the output in JSON format, delineating separate labels for entity type and intention. Sample sentences from the training set, along with their corresponding JSON output, were included as illustrative examples. This setup varied in the number and order of displayed examples based on the retrieval method, exploring the impact of these factors on model performance through separate experiments.

Attributive NER For Attributive NER, the prompt construction mirrored the approach taken in Subtask 1 but incorporated more detailed rules, emphasizing the necessity for attributes to relate directly to software entities. Known software entities were additionally provided as input, even for test instances, adhering to the oracle setup defined in the shared task guidelines. This method aimed to refine the model's ability to extract and classify attributive information accurately.

Relation Extraction The approach to Relation Extraction reimagined the task as a series of single-choice question-answering challenges. Each potential relation, given the software and attributive entities within a sentence, was listed, with claims formulated

for each possible relationship (e.g., "IBM is the developer of Windows"). The model was tasked with identifying the veracity of each claim through a single-choice question format, where all claims were enumerated and solutions provided in a batch format for each example sentence. This setup culminated in presenting the test instance alongside its single-choice questions, expecting the model to deliver decisions on the relational claims.

4.5.3 Train Sample Retrieval for Few-Shot Generation

Our experimental framework explored various methods for test sample retrieval to ascertain the most effective approach in enhancing model performance. These methods included the use of random illustrative samples to represent every possible signature, retrieval based on entity similarity, and retrieval based on sentence similarity. Each method was evaluated against a baseline to determine its impact on the accuracy and efficiency of the information extraction tasks at hand.

4.5.4 Relation Extraction Baseline

In the development of our baseline for relation extraction, we established a robust heuristic framework derived from an analysis of potential relations indicated within the text. Our strategy was guided by two principal rules aimed at simplifying the decision-making process for identifying accurate relationships between entities. Firstly, we limited our consideration to relations that necessitate the presence of at least one related entity, deliberately excluding optional inter-software entity relations such as "specification_of" and "PlugIn_of". Given the infrequent occurrence of these cases within the training dataset, we anticipated only a minimal impact on the overall performance metric, specifically the weighted mean macro F1 score for the subtask. Secondly, for all remaining relation types, our approach favored selecting the closest possible entity positioned to the left of the focal software entity as the most likely relation partner. This heuristic was not only straightforward but proved to be highly effective, aligning our baseline performance with that of the top contenders in the shared task. This methodology highlights the potential of leveraging simple, rule-based strategies to achieve competitive results in complex relation extraction challenges.

Paradigm	**Retrieval**	**F1**	**Model**	**Parameter**
Finetuned	-	0.599	SciBERT	-
GLM	Random	0.483	GPT 3-5	Random k=7
GLM	Random	0.525	GPT 3-5	Random all entity types shown
GLM	Sim. sentences	0.647	GPT 3.5	topn=10
GLM	Sim. entities	0.624	GPT 3.5	topn=7 x n entities
GLM	Random	0.574	GPT 4	Random all entity types shown
GLM	Sim. sentences	0.677	GPT 4	topn=10
GLM	Sim. entities	**0.679**	GPT 4	topn=7 x n entities

Table 4.1: Results of Different Models and Retrieval Methods on Subtask 1

4.6 Results

Our analysis in Subtask 1 (Software NER) shows a varied performance landscape across different models and retrieval methods (Table 4.1). The finetuned baseline, which uses SciBERT, achieved a solid foundation with a 59.9% F1 score. However, GLMs that use random samples without fine-tuning showed a decrease in performance, with the highest F1 score reaching only 57.4%.

Task	Rank	User	F1	Precision	Recall
Subtask 1: Software NER					
1	1	**phinx**	**.740**	**.761**	**.750**
1	2	david-s477	.692	.739	.711
1	3	ThuyNT03	.678	.729	.649
1	4	*ours*	*.652*	*.679*	*.664*
1	5	vampire	.648	.682	.637
Subtask 2: Attribute NER					
2	1	***ours***	***.838***	***.835***	***.847***
2	2	phinx	.743	.745	.748
Subtask 3: Relation Extraction					
3	1	***ours***	***.916***	*.911*	***.924***
3	2	phinx	.897	.900	.897
3	–	*baseline*	*.864*	*.857*	*.875*
3	–	*necessary*	*.562*	***.933***	*.415*

Table 4.2: SOMD Performance Rankings (Weighted Average Macro).

A closer examination of retrieval-based models indicates that GLMs perform better. The highest F1 score of 67.9% was achieved by sentence similarity retrieval models, while entity retrieval showed the best performance at 67.7% F1. The transition from GPT 3.5 to GPT 4 models resulted in a significant improvement of approximately 3–5%, although it required around three times more computation time. Notably, our best models were able to perform within a mere 3% below the theoretical maximum by utilizing SciBERT for sentence selection, leveraging oracle positive sentences. In Subtask 2 (Attributive NER), our methodology showed a significant improvement of +10% in F1 performance compared to our competitors, demonstrating the effectiveness of our approach in a low data regime (Table 4.2). For Subtask 3 (Relation Extraction), our GLM Single-Question Answering model further improved the F1 score by 5.1%, building on the already competent performance of the heuristic baseline and highlighting the advantage of our method. Furthermore, it has been demonstrated that using 7-10 samples is the most effective strategy, as it optimises the balance between input complexity and model performance.

This analysis highlights the potential of utilising advanced GLM techniques and carefully selected retrieval methods to significantly improve the accuracy and efficiency of NER tasks in specialised domains.

4.7 Conclusion

Our research on enhancing Relation Extraction (RE) with Generative Language Models (GLMs) through Single-Choice QA has introduced a novel intersection of methodologies aimed at improving the precision of information extraction in the context of scientific texts. By integrating Retrieval-Augmented Generation (RAG) with GLMs and adopting a methodical approach to fine-tuning and leveraging large language models such as SciBERT and use the model to support GPT variants, we have demonstrated the capability of GLMs to navigate the complexities inherent in the extraction of software entities and their attributes.

The exploration of different retrieval strategies—ranging using entity and sentence similarity underscores our commitment to refining the inputs for generative models, ensuring that they are fed the most relevant and contextually appropriate data. This meticulous preparation has allowed us to significantly boost the performance of GLMs in recognising nuanced distinctions among software-related entities and accurately extracting relation types within scholarly articles. Our experiments have not only highlighted the efficacy of GLMs in addressing domain-specific tasks with a limited set of examples but also revealed the inherent challenges, such as the difficulties in matching entity mentions accurately. Despite these hurdles, our single-choice QA approach for RE lead to a strong heuristic baseline for relation extraction and show how altering the task simplifies the problem.

The outcomes of our research indicate a promising direction for future work in leveraging GLMs for NER and RE tasks. The development of our system for participation in the SOMD shared task has illustrated the potential of a single-choice QA approach to relation extraction, offering a structured and scalable method for extracting meaningful insights from textual data. Our findings contribute to the growing body of knowledge on the application of generative models in the field of computational linguistics, paving the way for more sophisticated and efficient methodologies in information extraction from scientific literature.

4.8 Chapter Appendix

4.8.1 Dataset Overview

Each subtask in the SOMD Shared Task is evaluated independently. Subtask 2's attributive NER uses the identified software entities from Subtask 1 as inputs, and the annotations from the first two subtasks are combined for Subtask 3's Relation Extraction. The dataset is hierarchically structured, with each subtask's test set forming a subset of the previous one. This lead to data leakage in Subtask 1 and to a lesser extent in Subtask 2. This design ensures that tasks are interconnected, but it requires careful interpretation of results, particularly with respect to error propagation. Although this design creates data leakage, it ensures the interconnectedness and cumulative nature of the challenges. The initial presentation of the Corpus is as the SoMeSci knowledge graph (Schindler et al., 2021). It is a comprehensive dataset consisting of 1,367 documents and containing 399,942 triples that represent 47,524 sentences. The dataset includes 2,728 software entities, totaling 7,237 labeled entities. The dataset includes positive and negative examples, i.e. sentences without software mentions, which

improves the accuracy of the model. The presence of duplicate sentences, headings, and varying sentence lengths increases the complexity of the text and makes extraction tasks from scientific texts more challenging.

Challenges in Applying GLMs to NER Tasks

The integration of Generative Language Models (GLMs) into Named Entity Recognition (NER) tasks introduces a set of unique challenges that can significantly impact the performance and reliability of extraction outcomes. One of the most prevalent issues in generative NER approaches is the phenomenon of hallucination (S. Wang et al., 2025), where the model generates entities not present in the test instance. This can result from the model misinterpreting the provided examples as part of the text from which entities should be extracted, leading to inaccuracies and inconsistencies in the results.

Furthermore, matching problems during the location of mention positions present considerable challenges, particularly in the context of span-based evaluation systems. These systems evaluate the accuracy of entity extraction based on the precise span of text identified as an entity. Discrepancies in the extracted span—whether through corrected spellings or variations in representation—can complicate the matching process. For example, the extraction of "jquery" as "jQuery" illustrates a common scenario where the prevalent spelling of a software library may differ from its mention in the text, yet both are correct. This variability necessitates sophisticated matching strategies to ensure accuracy in evaluation.

The situation is further complicated by texts containing multiple mentions of a single entity. The challenge lies in determining whether each mention can be accurately classified as an entity, alongside dealing with the ambiguity of different entities that are written in the same way. Overlapping or nested entity mentions that do not align with the ground truth data introduce additional layers of complexity, requiring nuanced approaches to entity recognition and classification. Our proposed baseline solution, filtering out non-matching entities and employing rule-based decisions for handling multiple matches have been adopted. However, the disadvantage of this method is that it is based on simplistic heuristics.

Future work may explore advanced solutions like using GLMs for precise entity matching, offering potential improvements for NER challenges. While this paper does not delve into these complex methods, it highlights the importance of ongoing research to further refine and improve GLMs for more accurate and efficient entity extraction.

Subtask 3: Domain and Ranges for Train Set Properties

Range Class ⇒ Object Property	**Application**	**Operating System**	**PlugIn**	**Programming Environment**
Abbreviation ⇒ Abbreviation_of	✓	✓	✓	✓
Developer ⇒ Developer_of	✓	✓	✓	✓
Release ⇒ Release_of	✓	✓	✓	✓
Version ⇒ Version_of	✓	✓	✓	✓
Citation ⇒ Citation_of	✓	×	✓	✓
PlugIn ⇒ PlugIn_of	✓	×	✓	✓
URL ⇒ URL_of	✓	×	✓	✓
Extension ⇒ Extension_of	✓	✓	×	×
License ⇒ License_of	✓	×	✓	×
AlternativeName ⇒ Alternative-Name_of	✓	×	×	×
Application ⇒ PlugIn_of ⇒ Specification_of	✓	×	×	×
ProgrammingEnvironment ⇒ Specification_of	×	×	×	✓

Table 4.3: Object Properties, Domain, and Range Classes with Inner Software Relations

4.8.2 Similarity Search Examples

Search By Entity Similarity

Entity	Label	Sim.	Split	Sentence
PhosphOrtholog	Application Creation	1.00	test	To this end , we have developed an automated web - based tool , PhosphOrtholog , which allows batch processing and mapping of large species - specific PTM datasets to compare overlap at a site - specific level .
SNPdetector	Application Creation	0.93	train	We developed a software tool , SNPdetector , for automated identification of SNPs and mutations in fluorescence - based resequencing reads .
ESPRIT-Forest	Application Creation	0.92	train	In this paper , we developed a new algorithm called ESPRIT - Forest for parallel hierarchical clustering of sequences .

Table 4.4: Retrieval example based on entity similarity for RAG. The first row is the query entity. "sim" reflects the cosine similarity of the entities.

Search By Sentence Similarity

Split	Sim.	Text
test	1.00	To this end , we have developed an automated web-based tool , PhosphOrtholog , which allows batch processing and mapping of large species-specific PTM datasets to compare overlap at a site-specific level .
train	0.99	Here we present a tool , Podbat (Positioning database and analysis tool) , that incorporates data from various sources and allows detailed dissection of the entire range of chromatin modifications simultaneously .
train	0.99	We designed and developed a new method , MSACompro, to synergistically incorporate predicted secondary structure , relative solvent accessibility , and residue – residue contact information into the currently most accurate posterior probability - based MSA methods to improve the accuracy of mult ...
train	0.99	In this paper , we present a lossless compression tool , MAFCO, specifically designed to compress MAF (Multiple Alignment Format) files .

Table 4.5: Retrieval example based on sentence similarity for RAG. The first row is the query entity. "sim" reflects the cosine similarity of the sentences. The example sentence is the same as in Table 4.4

4.8.3 Prompting Examples

Prompt 4.1: Software NER Few-Shot prompt (n=2). The shown sample is the same as in the similarity examples in Table 4.4 and 4.5.

```
# Task: Extract entities in a closed NER setting and classify their intention.

## Entity Types (closed setting):
The entity types in this closed setting are: ['Application', 'ProgrammingEnvironment', '
    SoftwarePackageOrPlugin', 'OperatingSystem', 'SoftwareCoreference']
Do not include version information for the entities.
if no entity of the given entity types in the context return an empty list

## Intention Classification:
For each entity mention classify the intention of the usage in the context.
Use only and exact one intention per entity.
The intention classes are: ['Usage', 'Deposition', 'Creation', 'Mention']

## The Contexts:
The contexts to extract the entities are about software and from scientific publications.

## Output Format:
Return the result in a json list of objects.

# Examples:

## Context:
"""We␣developed␣a␣software␣tool␣,␣SNPdetector␣,␣for␣automated␣identification␣of␣SNPs␣and␣mutations␣
    in␣fluorescence␣-␣based␣resequencing␣reads␣."""
## Detected Named Entities with Intention:
[
  {
    "text": "SNPdetector",
    "label": "Application",
    "intention": "Creation"
  }
]
## Context:
"""In␣this␣paper␣we␣developed␣a␣new␣algorithm␣called␣ESPRIT␣-␣Forest␣for␣parallel␣hierarchical␣
    clustering␣of␣sequences␣."""
## Detected Named Entities with Intention:
[
  {
    "text": "ESPRIT␣-␣Forest",
    "label": "Application",
    "intention": "Creation"
  }
]
## Context:
"""To␣this␣end␣,␣we␣have␣developed␣an␣automated␣web␣-␣based␣tool␣,␣PhosphOrtholog␣,␣which␣allows␣
    batch␣processing␣and␣mapping␣of␣large␣species␣-␣specific␣PTM␣datasets␣to␣compare␣overlap␣at␣a␣
    site␣-␣specific␣level␣."""
## Detected Named Entities with Intention:
```

Prompt 4.2: Attributive NER Few-Shot prompt (n=2).

```
# Task:
## Task Description
Extract attributive text spans about software related entities in a closed attibutive NER setting.
## The Contexts:
The contexts to extract the attributive text spans are about software and from scientific
    publications.
## The Software Related Entities:
The software related entities are given as a list of json object with mention and entity type.
Do not annotate any related software entities.
## Attribute Types (closed setting):
The attribute types to identify in this closed setting are:
```

```
 ['Version', 'Developer', 'Citation', 'URL', 'Release', 'Abbreviation', 'License', 'Extension', '
     AlternativeName']
Keep the "Version" note separate, without overlap with other entities or attribute spans.
No span overlaps are allowed inbetween attributive spans of to the given software related entites.
Note that the attribute types must be associated with a software-related entity of one of the
    following types: ['Application', 'ProgrammingEnvironment', 'PlugIn', 'OperatingSystem', '
    SoftwareCoreference'].
To help you, you will receive all software-related entities as json for each context.
## Output Format:
Return the result in a json list of objects.
# Examples:
## Context:
"""LAMINA␣uses␣the␣Java␣Advanced␣Imaging␣(␣JAI␣)␣package␣http://java.sun.com/javase/technologies/
    desktop/media/jai/␣to␣support␣common␣image␣file␣formats␣,␣which␣is␣bundled␣with␣the␣
    installation␣and␣hence␣no␣additional␣installation␣should␣be␣required␣."""
### Detected Software Related Named Entities:
[
  {
    "text": "LAMINA",
    "label": "Application"
  },
  {
    "text": "Java␣Advanced␣Imaging",
    "label": "Application"
  }
]
### Detected Attributive Text Spans:
[
  {
    "text": "JAI",
    "label": "Abbreviation"
  },
  {
    "text": "http://java.sun.com/javase/technologies/desktop/media/jai/",
    "label": "URL"
  }
]
## Context:
"""The␣library␣,␣called␣ENCORE␣(␣http://encore-similarity.github.io/encore␣)␣,␣interfaces␣with␣the␣
    MDAnalysis␣molecular␣analysis␣toolkit␣[14]␣and␣can␣be␣used␣both␣as␣a␣Python␣library␣and␣from␣
    the␣command␣line␣."""
### Detected Software Related Named Entities:
[
  {
    "text": "ENCORE",
    "label": "PlugIn"
  },
  {
    "text": "MDAnalysis␣molecular␣analysis␣toolkit",
    "label": "PlugIn"
  },
  {
    "text": "Python",
    "label": "ProgrammingEnvironment"
  }
]
### Detected Attributive Text Spans:
[
  {
    "text": "http://encore-similarity.github.io/encore",
    "label": "URL"
  },
  {
    "text": "[14]",
    "label": "Citation"
  }
]
## Context:
```

```
"""A␣Virtual␣Machine␣for␣Oracle␣'␣s␣VirtualBox␣has␣also␣been␣built␣to␣provide␣easy␣access␣to␣IDEPI␣
    for␣users␣unfamiliar␣with␣the␣intricacies␣of␣Python␣package␣management␣,␣and␣is␣available␣from␣
    the␣main␣package␣distribution␣page␣(␣http://github.com/veg/idepi/␣)␣."""
### Detected Software Related Named Entities:
[
  {
    "text": "VirtualBox",
    "label": "Application"
  },
  {
    "text": "IDEPI",
    "label": "PlugIn"
  },
  {
    "text": "Python",
    "label": "ProgrammingEnvironment"
  }
]
### Detected Attributive Text Spans:
```

Prompt 4.3: Single-choice QA prompt for relation annotation.

```
# Answer the following single-choice questions on sentences from scientific publications on the
    subject of software.
 * Please note that there is exactly one answer to each question.

## Sentence:
"""The␣web␣service␣[44]␣available␣at␣http://www.fragit.org␣enables␣users␣to␣upload␣their␣structure␣,
    ␣fragment␣it␣and␣download␣the␣resulting␣input␣file␣to␣GAMESS␣."""

### Question(s):

Which of the statements about the Citation '[44]' is true?
idx_0: '[44]' is the citation of 'web␣service'.
idx_1: '[44]' is the citation of 'GAMESS'.

Which of the statements about the URL 'http://www.fragit.org' is true?
idx_2: 'http://www.fragit.org' is the url of 'web␣service'.
idx_3: 'http://www.fragit.org' is the url of 'GAMESS'.

### Answers to the questions about the sentence(s):
idx_0: True
idx_1: False

idx_2: True
idx_3: False

## Sentence:
"""ASPASIA␣,␣released␣under␣the␣Artistic␣License␣(␣2.0␣)␣,␣can␣be␣downloaded␣from␣http://www.york.ac.
    uk/ycil/software␣."""

### Question(s):

Which of the statements about the Version '2.0' is true?
idx_4: '2.0' is the version of 'ASPASIA'.
idx_5: '2.0' is the version of 'Artistic␣License'.

Which of the statements about the URL 'http://www.york.ac.uk/ycil/software' is true?
idx_6: 'http://www.york.ac.uk/ycil/software' is the url of 'ASPASIA'.
idx_7: 'http://www.york.ac.uk/ycil/software' is the url of 'Artistic␣License'.

### Answers to the questions about the sentence(s):
idx_4: False
idx_5: True

idx_6: True
idx_7: False

```

```
## Sentence:
"""The␣web␣-␣tool␣can␣be␣accessed␣from␣http://www.phosphortholog.com/␣using␣any␣typical␣web␣browser␣
    (␣except␣Internet␣Explorer␣)␣."""

### Question(s):

Which of the statements about the URL 'http://www.phosphortholog.com/' is true?
idx_8: 'http://www.phosphortholog.com/' is the url of 'web␣-␣tool'.
idx_9: 'http://www.phosphortholog.com/' is the url of 'Internet␣Explorer'.

### Answers to the questions about the sentence(s):
```

5

Modeling Relations between Machine Learning Artifacts

Publication Information

Title: GSAP-ERE: Fine-grained Scholarly Entity and Relation Extraction focused on Machine Learning

Authors: Wolfgang Otto, Sharmila Upadhyaya, Lu Gan, Saurav Karmakar, Stefan Dietze

Status: Published AAAI 2026 full paper (Archival, Peer-Reviewed)

Reuse: This chapter reproduces the text of the publication listed above in thesis layout.

Context

This publication extends the thesis from identifying entities to modeling the relations between them. While GSAP-NER establishes a fine-grained inventory of ML-related entities, many research questions require knowing how these entities are connected, for example which datasets were used for training or evaluation, which methods are applied to which tasks, or which models are compared. GSAP-ERE addresses this by adding relation annotations on top of the GSAP entity schema and by defining a relation set that captures key links between methods, models, tasks, datasets, and related mentions.

The chapter contributes a manually annotated corpus and an annotation scheme that is tailored to common reporting patterns in ML papers, including informal mentions where they help represent the text as written. This provides a stronger basis for extraction pipelines that aim at structured representations of experimental setups and claims. In addition to the dataset contribution, the paper evaluates baseline approaches for entity and relation extraction, including supervised models and prompting-based LLM pipelines. This creates a shared empirical reference point for later comparisons.

Within the thesis structure, GSAP-ERE is the step that turns entity-level extractions into structured statements that can populate knowledge graphs or support analysis of ML research practices. It directly feeds into the cross-dataset alignment and evaluation work in the next chapter, and it defines relation types that are relevant for later system-level exploration and user-facing applications.

Abstract

Research in Machine Learning (ML) and AI evolves rapidly. Information Extraction (IE) from scientific publications enables to identify information about research concepts and resources on a large scale and therefore is a pathway to improve understanding and reproducibility of ML-related research. To training and testing of IE models focused on fine-grained information in ML-related research, e.g. method training and data usage, we introduce GSAP-ERE. It is a manually curated fine-grained dataset of mentions of 63K ML-related entities and 35K relations distributed across 10 entity types and 18 semantically categorized relation types annoated in the full text of 100 ML publications. We show that our dataset enables fine-tuned models to automatically extract ML-related information that facilitate knowledge graph (KG) construction from scholarly papers or monitoring of computational reproducibility of AI research at scale. Additionally, we use our dataset as a test suite to explore prompting strategies for IE using Large Language Models (LLM). We observe that the performance of state-of-the-art LLM prompting methods is largely outperformed by our best fine-tuned baseline model (NER: 80.6%, RE: 54.0% for the fine-tuned model vs. NER: 44.4%, RE: 10.1% for the LLM). This disparity of performance between supervised models and unsupervised usage of LLMs suggests datasets like GSAP-ERE are needed to advance research in the domain of scholarly information extraction.

5.1 Introduction

Scholarly information extraction facilitates mining and understanding of vast amounts of scholarly literature (Luan et al., 2018; Luan et al., 2019; Jain et al., 2020; Pan et al., 2023; Q. Zhang et al., 2024) and usually involves Named Entity Recognition (NER) and Relation Extraction (RE), enabling the identification of entities and their relationships. The joined Entity and Relation Extraction (ERE) enables tasks, such as academic question answering (Dasigi et al., 2021) and scholarly knowledge graph construction (Viswanathan et al., 2021), and thus facilitates reproducibility and reusability of research.

In particular among scholarly entities relevant to AI and ML-related research, e.g., dataset, ML model and task, various dependencies are of great interest to enhance research reproducibility and reusability (Arvan et al., 2022; Wonsil et al., 2023). *Data-model dependencies* describe the datasets used for training and evaluating models; *model-task dependencies* indicate the downstream tasks applied to the trained or fine-tuned models; *inner-entity-type dependencies* capture comparability between scholarly resources or research concepts; and *entity-tracing dependencies* reflect the provenance or source of research artifacts.

However, many existing IE methods build upon datasets with a coarse-grained entity label set (Luan et al., 2018; Q. Zhang et al., 2024) that are not able to distinguish fine-grain artifact metadata. Second, despite the huge progress in recent years, vanilla LLMs (A. Yang et al., 2025; Touvron et al., 2023) still lag behind largely on fine-grained domain-specific IE tasks compared to fine-tuned models (Q. Zhang et al., 2024). Hence, it is not suitable to directly apply vanilla LLMs for a high-quality fine-grained scholarly IE. Third, despite of initial efforts to create a fine-grained scholarly NER dataset like GSAP-NER (Otto et al., 2023) or an RE dataset with wide relation coverage

	ScienceIE	SemEval18 Task 7	SciERC	SciER	GSAP-ERE
Annotation Unit	♣	♠	♠	♥	♥
# Publications	500	500	500	106	100
# Entity Types	3	-	6	3	10
# Relation Types	2	6	7	9	18
# Entities	9,946	7,505	8,094	24,518	62,619
# Relations	672	1,583	4,648	12,083	35,302
# Entities in Relations	1,824 (18%)	3,166 (42%)	6,272 (77%)	17,148 (70%)	46,680 (75%)
# entity Sentences	3,319	2,403	2,551	7,722	20,020
# null Sentences	654	153	136	2	6185
# Relations/Pub.	3.1	3.3	9.3	114.0	353.0
# Sentences/Pub.	8.1	5.1	5.4	72.9	262.0

Table 5.1: Comparison of GSAP-ERE and four datasets supporting NER and RE in scientific text. Note that our dataset is the biggest dataset in terms of annotated entities and relations so far and includes checked *null Sentences* without entity mentions as negative samples. Annotation units: ♣=Paragraph, ♠=Abstract, ♥=Full Text.

like SciER (Q. Zhang et al., 2024), a comprehensive fine-grained entity and relation extraction dataset is still lacking.

We therefore introduce GSAP-ERE, a comprehensive dataset of fine-grained entity and relation types, based on high-quality expert annotations at span level focused on ten entity types important in ML research.
Our contributions include:

- A reusable data model that captures the complex relations among ML models, datasets and tasks. We introduce a set of 18 relation labels, which are systematically organized into seven distinct semantic relation groups. We provide the data model visualization in the extended version.

- We provide a dataset of more than 63K entity mentions and 35K fine-grain relation annotations in 100 scientific publications, encompassing connections among ML models, methods, datasets, and tasks. Our annotations further include additional links to references and URLs, and cover all sentences in the full text of the selected publications, including those without relations or entities defined by our data model (null cases).

- To demonstrate the utility of our dataset, we present fine-tuned baseline models that showcase the applicability of automatic entity and relation extraction using both pipeline and joint modeling approaches. We also report zero-shot and few-shot performance results for unsupervised prompting approaches with large language models (LLMs).

The dataset, code for replicating the baseline models, and the annotation guideline are provided on our project site https://data.gesis.org/gsap/gsap-ere.

5.2 Related Work

Scholarly IE Datasets Multiple datasets (Augenstein et al., 2017; Gábor et al., 2018; Luan et al., 2018; Schindler et al., 2021; Q. Zhang et al., 2024), have been proposed as ground-truth datasets for scholarly information extraction tasks, including Named Entity Recognition (NER) and Relation Extraction (RE). We compare our GSAP-ERE with four well-established, manually annotated ground-truth datasets for scholarly IE in Table 5.1.

ScienceIE at SemEval 2017 (Augenstein et al., 2017) and SemEval 2018 Task 7 (Gábor et al., 2018) are two pioneering ground-truth datasets for scholarly IE that contain both entity and relation annotations. ScienceIE includes 500 paragraphs from open access journals and contains three entity types: *Task*, *Method* and *Material* and two relation types: *Hyponym-Of* and *Synonym-Of*. SemEval 2018 Task 7 provides six types of relations on annotation of entity span without typing based on abstracts from NLP publications. Another dataset SciERC (Luan et al., 2018) has been proposed, extending ScienceIE and SemEval 2018, where 500 publication abstracts from 12 AI conference or workshop proceedings are annotated. SciERC contains six entity types, including one type *Generic* supporting informal mentions of entities, and seven relation types. However, all these datasets are only based on abstracts (Gábor et al., 2018; Luan et al., 2018) or selected paragraphs (Augenstein et al., 2017) of publications, rather than full texts, which can cover more diverse linguistic styles (Q. Zhang et al., 2024) and thus potentially render a wider range of entities and relations.

Therefore, full-text-based ground-truth datasets are introduced. DMDD (Pan et al., 2023) is a dataset automatically annotated with distant supervision on the full text of 31 219 scientific articles. Yet DMDD does not contain relation annotations, hence is not suitable for the relation extraction task. SciREX (Jain et al., 2020) contains four entity types but its relation annotations are limited to clusters of mentions rather than individual entity mention pairs, overlooking the contextual information of each entity mention. SoMeSci (Schindler et al., 2021) contains the full text of 100 articles from PubMed Central Open Access subset; however it focuses on various software mentions rather than ML-related entity types. SciER (Q. Zhang et al., 2024), deriving from the NER dataset SciDMT (Pan et al., 2024), provides a corpus of 106 full-text annotated publications with three entity types (*Dataset*, *Task*, *Method*) and ten relation types. To our best knowledge, SciER has the most comprehensive relation annotation scheme among the existing comparable datasets. However, it only includes explicitly named mentions, excluding the unnamed or informal mentions of entities, which accounts for a substantial amount of mentions in research publications, and results in an incomplete set of relation annotations. Additionally, the authors do only provide annotated sentences, resulting in only 73 sentences per document in the training and test data.

Compared to these datasets, our dataset, derived from GSAP-NER (Otto et al., 2023), is based on a manual annotation of 100 full-text research publications in the fields of ML and applied ML. It contains 10 fine-grained entity types and 18 relation types. The entity label set includes both named and informal entity types, in contrast to SciER. To enable a wide range of downstream use cases, the 18 relation types cover 7 different semantic groups relating to the creation, usage, property, application and referencing of scholarly entities. Furthermore, the co-existence of flat, nested and overlapping entity mentions renders it an even more challenging relation extraction task.

Semantic Group	Purpose	Example
Model Design	Captures internal structure and dependencies of methods and ML models.	*architecture* → **BART** follows a standard **sequence-to-sequence Transformer** architecture.
Task Binding	Links models and datasets to the associated tasks.	*appliedTo* ← For **NER**, we consider **CoNLL-2002** (Sang, 2002) and CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003) datasets ...
Data Usage	Describes how datasets are used in methods and models.	*evaluatedOn* → To that end, we evaluate **XLM-R** on the **GLUE** benchmark.
Data Provenance	Tracks the origin and transformation of datasets.	*sourcedFrom* ← **PubMed** alone has a total of **29M articles** as of January 2019.
Data Properties	Encodes metadata of datasets (size and instance type).	*size* → **GOOAQ** contains **3 million questions** with short, snippet, or collection answers, such as the ones shown in Fig. 1.
Peer Relations	Covers semantic, logical and structural relation among same entity type.	*isComparedTo* Comparing to **WIKISQL-WEAK**, **WIKITABLE-QUESTIONS** requires more complicated reasoning capabilities.
Referencing	Refers to external sources or documentation (citation and URL).	*citation* → The latter is inspired by **TAPAS** (**Herzig et al., 2020**) ...

Table 5.2: Brief definition of each semantic relation group and a corresponding example from our GSAP-ERE dataset. Note that each example shows only one relation annotation that represents the corresponding semantic group. Other entity mentions and relations in the gold data are not shown. The highlighted entity types (**MLModel**, **ModelArchitecture**, **Task**, **Dataset**, **DatasetGeneric**, **DataSource**, **ReferenceLink**) match the colors used in the examples.

Entity and Relation Extraction Entity and relation extraction, aiming at extracting relations from text, is composed of two sub-tasks: NER and RE tasks. To achieve the end-to-end relation extraction, there exists two major paradigms: the pipeline-based approaches (Zhong and Chen, 2021; D. Ye et al., 2022) and the joined extraction approaches (Luan et al., 2019; Yan et al., 2023). PURE (Zhong and Chen, 2021) is an pipeline-based approach that uses two pre-trained encoders for the NER and RE models to capture the contextual information for each span or span pair. It also proposes an approximate batched relation model that shares the position embedding between the entity marker and the entity start and end tokens to accelerate the inference. `PL-Marker` (D. Ye et al., 2022) is another pipeline-based approach, extending PURE batched relation model with packed levitated marker to model entity pair interaction. `HGERE` (Yan et al., 2023) is a state-of-the-art joint extraction approach back-boned by the PL-Marker framework (D. Ye et al., 2022) and incorporates a hypergraph neural network to facilitate information flow among entities and relations. We fine-tune and evaluate both state-of-the-art pipeline and joint ERE models to compare their performance on GSAP-ERE.

5.3 Preliminaries

We formalize the entity and relation extraction (ERE) task at the text span level, enabling the tracing of extracted entities and relations back to the specific sentences from which they originate. Formally, we define two sub-tasks, NER and RE, as follows. We consider a collection of publications D as input. Each publication is divided into paragraphs $P = \{p_1, p_2, \ldots, p_n\}$, with each paragraph further segmented into consecutive sentences $S = \{s_1, s_2, \ldots, s_m\}$. For each tokenized sentence $s_i = \{w_1, w_2, \ldots, w_n\}$, we define two extraction steps to identify entities and relations as follows.

NER We define the task of NER with a closed set of entity labels L^{NER} as the identification of contiguous word sequences $e_i = \{w_l, \ldots, w_r\}$ (i.e., spans) with a corresponding entity label from L^{NER}.

RE In the second step, we define a closed RE task using a label set L^{RE}, which includes a NIL label to denote the absence of a relation. For every possible pair of entities $E^p = \{(e_i, e_j) | i \neq j\}$ we define the task of RE as the assignment of all pairs to one of the labels L^{RE}.

ERE is the joined task of NER and RE from end-to-end independent of the design of the used approaches.

Metrics A joint evaluation, consistent with the task definition of ERE, encompasses both entity and relation checking. For NER task, exact match (***NER***) and partial match (***NER***≈) are two common evaluation settings. For RE, we note that its difficulty is substantially higher than that of NER. This is due to the fact that any relation pair includes two entity mentions with a relation label, where both entity mentions and relation label can lead to misalignment. Therefore, to support the assessment of relation annotations, we introduce four evaluation settings as below.

RE+ refers to both the correct labels for the entities and the relation, and exact span matching for the entities;

RE asks only correct relation labels and exact entity spans, with no restriction for the entity types;

RE+≈ requires exact match of all labels but allows partially overlap of entity spans;

RE≈ only asks the exact relation label and overlapped entity spans, with no further requirement.

We report micro F1 metrics, unless otherwise specified.

5.4 GSAP-ERE Dataset

In this section, we present the detailed curation process of our GSAP-ERE dataset. We reuse the comprehensive entity label sets of GSAP-NER (Otto et al., 2023) and their entity annotation as a starting point to create relation annotations.

5.4.1 Data Collection and Processing

Since we extend and build upon the GSAP-NER dataset (Otto et al., 2023) with relation annotation, we first give a brief overview of the GSAP-NER dataset. GSAP-NER provides a corpus of 100 manually annotated full-text research publications in ML or applied ML fields. The publications are first sampled using a popularity-diversity combined strategy described in Otto et al., 2023. The publications are selected from a Hugging Face model search (Hugging Face, 2016) and arXiv metadata (Cornell University, 2022), and then converted from PDF to plain text format via GROBID (Lopez, 2008). The ten entity types in this corpus are categorized into three categories: (1) ML model related, i.e. *MLModel*, *ModelArchitecture*, *MLModelGeneric*, *Method* and *Task*; (2) dataset related, i.e. *Dataset*, *DatasetGeneric* and *DataSource*; (3) miscellaneous, including *ReferenceLink* and *URL*.

5.4.2 Data Annotation

5.4.2.1 Annotation Scheme

We aim at a fine-grained relation label set that can reflect the comprehensive interactions among ML models, datasets, tasks, methods, model architectures and the miscellaneous entities. The interaction information should serve to improve research reproducibility and reusability, covering the provenance, usage, transformation, application and comparison of the scholarly entities. Unlike SciER (Q. Zhang et al., 2024), we keep the informal mentions of named entities, corresponding to entity types *MLModelGeneric* and *DatasetGeneric*, in the annotation scheme following SciERC (Luan et al., 2018). We argue that the informal mentions not only capture valuable information for coreferential and compositional relations, but also increase the entity-per-sentence density to cover a larger amount of relation interactions and alleviate data sparsity for certain relation types. Therefore, we define a label set of 18 relation types for seven semantic groups.

We provide examples for each semantic group in Table 5.2 and list the domain and range information for each relation in our seven semantic groups in Table 5.3. Find examples for each relation in the extended version (Otto et al., 2025). More detailed definitions of relations are provided in the guideline on our project site.

5.4.2.2 Annotation Strategy

To ensure high annotation quality, we employ a two-phase annotation-refinement process. Similar to SciER (Q. Zhang et al., 2024) and GSAP-NER (Otto et al., 2023), we use the INCEpTION (Klie et al., 2018) platform.

During the initial annotation phase, annotations are conducted by two student annotators with computer science background, who have finished an annotation training. Among the 100 publications, we randomly selected 10 publications for double annotation by the two annotators separately; and the remaining 90 publications are split and assigned to a single annotator each. In this phase, the annotators annotate the relations on the assigned publications with existing entity annotations introduced by GSAP-NER (Otto et al., 2023). In addition, the annotators are asked to check the existing entity annotations first to ensure the relation annotations are based on correct entity annotations.

In the refinement phase, we have two PhD students and two postdoc researchers in computer science to inspect the annotation alignment of the two annotators. Misalignment cases are filtered out in the 10 commonly annotated documents by both annotators, and patterns of misalignment on each relation are extracted. Relations in jointly annotated documents that show such misalignment patterns are re-checked and corrected if necessary on all 100 publications.

Domain	Relation	Range
Model Design		
Method	usedFor	Method MLModel(Generic)
MLModel(Generic) Method	architecture	ModelArchitecture
MLModel(Generic)	isBasedOn	MLModel(Generic)
Task Binding		
MLModel(Generic) Method	appliedTo	Task
Dataset(Generic)	benchmarkFor	Task
Data Usage		
MLModel(Generic) Method	trainedOn evaluatedOn	Dataset(Generic)
Data Provenance		
Dataset(Generic)	transformedFrom	Dataset(Generic)
	generatedBy	Method
	sourcedFrom	DataSource
Data Properties		
Dataset(Generic)	size hasInstanceType	DatasetGeneric
Peer Relations		
Any	coreference isPartOf isHyponymOf isComparedTo	Same as Subject
Referencing		
Any	citation	ReferenceLink
	url	URL

Table 5.3: Domain and range information for all 18 relation types in our 7 semantic groups.

5.4.2.3 Human Agreement and Quality Enhancement

We calculate interrater agreement on the 10 jointly annotated publications. We note that relation annotation is highly influenced by the underlying entity annotation. A revision of entity annotations preceding the relation annotation improves the NER interrater agreement from reported 0.61 (Otto et al., 2023) to 0.82 macro-F1 score under *NER* setting and 0.69 to 0.86 under *NER*≈ setting. We report NER interrater agreement in the extended version. Furthermore, due to the increasing difficulty of relation annotation, we evaluate the relation annotation in four settings as described in Section 5.3. We show the interrater agreement per semantic group in Table 5.4. Despite achieving a good overall alignment, the various agreement levels in different semantic groups confirm the challenge of our relation annotation, which may be partially due to the poor reporting practices mentioned in Section 5.1.

Semantic Group	***RE+***	***RE***	***RE+≈***	***RE≈***
Model Design	38.4	40.0	45.9	49.3
Task Binding	40.6	45.6	44.1	50.0
Data Usage	61.3	62.8	64.8	67.5
Data Provenance	51.1	51.5	58.6	60.0
Data Properties	80.1	80.1	84.6	84.6
Peer Relation	57.1	61.4	59.3	65.2
Referencing	61.6	68.1	65.2	73.7
weighted all	53.7	58.1	56.9	62.8

Table 5.4: Interrater agreement across different relation groups measured by F1 scores (%, micro average) under different evaluation settings explained in Section 5.3.

5.5 Experiments

We use our dataset to evaluate model abilities to extract relational knowledge from scholarly publications according to our problem definition in Section 5.3.

5.5.1 Baselines

5.5.1.1 Supervised PLM-based Baselines

Supervised entity and relation extraction (ERE) using pretrained language models (PLMs) is demonstrating state-of-the-art performance in the domain of scholarly documents (Q. Zhang et al., 2024). In our experiments we compare a pipeline-based approach, i.e. a separate NER step followed by a RE classification of the extracted entities, to a joint approach which learns to predict and classify entity mention spans and relations between pairs of detected entity mentions simultaneously using a joint loss.

PL-Marker (D. Ye et al., 2022). This pipeline-based approach first performs a span classification (NER) task, and then uses the proposed Packed Levitated Marker to model and evaluates the interaction between entity mention pairs for RE.

HGERE (Yan et al., 2023). This state-of-the-art joint approach extends the PLMarker framework and incorporates a hypergraph neural network to facilitate a high-order span classification for NER and entity pair classification for RE in one step.

We fine-tune `PL-Marker` and `HGERE` on our training and validation sets and then report the performance of each fine-tuned models, denoted as $\texttt{PL-Marker}_{\text{GSAP-ERE}}$ and $\texttt{HGERE}_{\text{GSAP-ERE}}$, on our test set. More specifically, we introduce and optimize a loss weight scheme in the `HGERE` approach in combination with *ternary* information flow configuration for the Hypergraph Neural Network (HGNN). For $\texttt{HGERE}_{\text{GSAP-ERE}}$ we tuned the hyperparameters in a stepwise process. We first adjusted the learning rate and found that `2e-5` gave the best results (tested: `1e-4`, `5e-5`, `2e-5`, `1e-5`). After that, we varied the batch size and obtained the best value with `18` (tested: `10`, `14`, `18`, `22`). We then tested different loss weighting factors to balance the NER and relation losses. Since the NER part learned faster, we increased the relative weight of the relation loss using the scheme $(1 - L) \cdot \text{loss}_{\text{ner}} + L \cdot \text{loss}_{\text{rel}}$. The best setting was $L = 0.9$ (tested: 0.1, 0.25, 0.5, 0.75, 0.9, 0.95). Finally, we varied the number of epochs and chose 8 (tested: 6, 8, 10). More experimental settings can be found in the extended version (Otto et al., 2025).

ERE-method	Model	***NER***	***NER≈***	***RE***	***RE≈***	***RE+***	***RE+≈***
Supervised pipeline	$\text{PL-Marker}_{\text{GSAP}}$	72.6	77.7	41.4	46.2	36.3	39.9
Supervised joined loss	$\text{HGERE}_{\text{GSAP}}$	**80.6**	**85.8**	**54.0**	**59.8**	**46.9**	**51.3**
Unsupervised (LLM)	Qwen 2.5 32b	42.0	56.9	7.2	14.6	7.2	10.9
	Qwen 2.5 72b	**44.4**	**59.1**	**10.1**	**15.7**	**8.2**	**11.9**
	LLaMA 3.1 72b	40.5	55.0	6.4	9.6	5.7	7.8

Table 5.5: Comparison of test set F1 score (%, micro average) for our baseline approaches. Label-strict metrics are marked with "+"; span-partial metrics with "≈". LLM-based results for NER are obtained with $k = 10$ examples and RE with $k = 1$ example. For both tasks we used the best performing configurations using the *similar+diverse* example selection strategy.

5.5.1.2 LLM Prompting

We run zero-shot and few-shot prompts on two state-of-the-art open-source generative LLMs, i.e. `LLaMA 3.1` (Touvron et al., 2023) and `Qwen 2.5` (A. Yang et al., 2025). Our prompting pipeline for NER and RE follows a two-stage process comparable to the supervised pipeline approach. For each input sentence, we first extract entities. These entities are then used to form subject-object candidate pairs, which are passed to a second prompt to classify their relation label. For n predicted entities, we generate $n(n-1)$ entity pairs and classify each using an individual prompt. Example prompts and templates used in the experiment are provided in the Appendix of the extended version (Otto et al., 2025). Each prompt is composed of five structured sections: *Task Introduction*, *Label Definitions*, *Step-by-Step Instructions*, *Few-Shot Examples* and *Main Input*, i.e. the actual input sentence to annotate. This structured prompt pattern is found to largely improve few-shot learning performance by helping align output generation with task expectations (Madaan et al., 2023).

To select the best performing hyperparameters, we explore two key prompt design parameters: the number of few-shot examples, and the example selection strategy. We apply various hyperparameter settings to the validation set to pick the best performing configuration before applying it to the test set. For the example selection, we compare random selection (a fixed, randomly chosen set) to a dynamic retrieval-based strategy. The dynamic strategy can be one of:

similar: retrieving top-k similar sentences via cosine similarity

similar+diverse: combining similar with maximizing unique label re-ranking

For our experiments, all examples were chosen from the training set. We evaluate the *random* and *similar+diverse* strategies on the ERE task. Each result was obtained from a single run per prompt.

5.5.2 Experimental Setup

For the supervised approaches, we fork the implementations of D. Ye et al., 2022 (code: Natural Language Processing Lab at Tsinghua University, 2022) and Yan et al., 2023 (code: Zhaohui Yan, 2023) and document the steps required to setup and train the models. For the supervised methods, we use the scibert-scivocab-uncased as encoder (Beltagy et al., 2019) (model: AllenAI, 2019). All supervised models are tested and evaluated using 10% of the annotated publications as validation and 10% as test set leaving 80 annotated publications for training. We selected for the *ternary* information flow configuration for the Hypergraph Neural Network in `HGERE` through hyperparameter optimization. We also tuned learning rate, batch size, and loss-weighting. Details of the optimization processes are provided in our Appendix in Otto et al., 2025. We run our best performing model with 5 random seeds and report the mean performance including the standard deviation for this model.

To guarantee reproducibility for the unsupervised prompting approach, we set the temperature parameter for all LLMs to zero. To select the best performing hyperparameter for the unsupervised prompting approach, we evaluated hyperparameters using `Qwen2.5 32B` on the validation set. For the final model comparison, we used three open-source LLMs: `Qwen2.5 (32B and 72B)` (A. Yang et al., 2025), and `LLaMA 3.1 70B` (Touvron et al., 2023) on the test set. For efficient local inference, we accessed the quantized versions of `Qwen 2.5` (model : Ollama, 2025b) and `LLaMA 3.1 70B` (model: Ollama, 2025a cite) using the Ollama, 2024 framework. For few-shot sentence selection, we use `multi-qa-mpnet-base-cos-v1` (model: Sentence-transformers, 2025) from the Sentence-Transformers library (Reimers and Gurevych, 2019) for sentence cosine similarity.
Experiments were conducted on a server running Ubuntu 22.04.4 LTS with 2× Intel Xeon 2.1 GHz CPUs (48 cores, 96 threads), 1.4 TB RAM, and 8 GPUs (4× RTX 2080 Ti with 11 GB and 4× A40 with 48 GB). The system also includes a multi-tier storage setup with both SSD and HDD arrays, totaling over 40 TB.

5.5.3 Evaluation Metrics

To comprehensively evaluate the performance of our baselines, we employ a set of metrics that capture different aspects and strictness levels for NER and RE introduced in the Preliminaries section. In general, we report micro F1 performance to reflect overall model performance across labels (Harbecke et al., 2022). In specific evaluation settings, we report micro, macro, and weighted-averaged metrics to summarize performance for detailed comparability.

Example Selection	Match	Metric	0	1	2	5	10	20
random	*NER*	micro-F1	19.1	24.7	23.1	29.7	34.1	**34.4**
		macro-F1	23.3	27.0	26.5	28.6	**31.5**	29.7
		weighted-F1	16.1	22.0	20.3	28.4	33.5	**33.6**
	NER≈	micro-F1	33.0	41.8	37.1	50.1	**53.3**	50.3
		macro-F1	34.5	40.3	37.7	43.7	**46.0**	40.9
		weighted-F1	29.8	39.6	34.5	49.4	**53.1**	49.4
similar+diverse	*NER*	micro-F1	19.1	34.7	38.2	40.4	**40.9**	27.8
		macro-F1	23.3	34.3	37.6	39.6	**39.7**	21.2
		weighted-F1	16.1	34.0	37.9	40.6	**41.4**	28.0
	NER≈	micro-F1	33.0	53.8	56.7	58.1	**58.4**	39.4
		macro-F1	34.5	48.1	50.9	**52.1**	52.0	28.4
		weighted-F1	29.8	53.4	56.5	58.2	**58.6**	38.8

Table 5.6: NER performance for different numbers of k-shot examples and example selection strategies on Qwen2.5-32B. We report micro, macro, and weighted F1 (%) for exact matching (NER) and partial matching (NER≈) on the validation set.

k-Shot	RE≈	RE+≈	RE	RE+
0	16.8	11.7	8.9	6.5
1	20.4	**14.4**	10.7	8.0
2	**20.5**	**14.4**	**10.8**	**8.1**
5	19.9	14.0	10.3	7.6

Table 5.7: F1 scores (%, micro average) for the RE task on the validation set using Qwen2.5-32B.

5.6 Experimental Results

5.6.1 Baseline Comparison

Table 5.5 presents a comparative overview of the supervised pipeline, joined, and the unsupervised prompting approaches using LLMs. Our best supervised approach outperforms the LLM-based approaches by a large margin with an *NER* F1 performance of 80.6 ± 0.3% and *RE+* F1 of 46.9 ± 0.5%. We report the best results for each setting with optimized hyperparameters. Specifically, the LLM-based results are all based on *similar+diverse* strategy, with 10 examples for NER evaluations and 1 example for RE evaluations. We detail the hyperparameter selection experiments in Section 5.6.2.

We first observe a substantial performance advantage of supervised models over unsupervised LLM prompting approaches, ranging from 18.6–39.8% for *NER*/*NER*≈ to 28.1%–50% for all four RE settings. Comparing the two supervised PLM approaches, the joint approach `HGERE` outperforms the pipeline approach `PL-Marker` under every setting for both NER and RE, demonstrating the robustness of this start-of-the-art method. Finally, among LLM prompting results, `Qwen 72b` achieves the best performance for all settings. However, compared to the two supervised PLM approaches, the advantage of `Qwen 72B` over other LLMs is marginal, in particular for the RE task, marking more research effort is needed for LLMs on such a comprehensive fine-grained IE task. Our findings are consistent with similar observations reported in related work (Q. Zhang et al., 2024).
When comparing runtime between the supervised and unsupervised approaches, the PLM-based methods outperforms the LLM-based methods by a substantial margin. On the test corpus of 10 documents, inference with the LLM approach is 182 times slower than with the fine-tuned PLM on the same hardware described in Section 5.5.2 (4 minutes vs. 12 hours and 29 minutes), while training the PLM required 2 hours and 30 minutes for a single training run.

5.6.2 Impact of LLM Few-Shot Hyperparameters

For our unsupervised prompting approach we run our pipeline on the validation set under different configurations to pick the optimized number of few-shot examples (k) and example selection strategy. Table 5.6 shows the results on NER in these configurations.[1] We observe that, consistently, the best performing configuration for NER is $k = 10$ examples with *similar+diverse* example selection, achieving 58% F1 score, +5% compared to random example selection. The performance sharply declines after 20 examples. For RE, we run $k \in \{0, 1, 2, 5\}$ with *similar+diverse* strategy. As shown in Table 5.7, the overall RE performance is unsatisfying, but 1-shot and 2-shot achieve the best performance with minimal performance differences.Taking the trade-off between extraction performance and computational costs into consideration, we select 10-shot *similar+diverse* for NER and 1-shot *similar+diverse* for RE as the optimized hyperparameters. The detailed results on the test set for pipeline-based NER and RE are provided in the extended version.

[1]We include the detailed performance in our extended version for further fine grained analysis.

5.7 Conclusion

We introduced GSAP-ERE, a manually annotated dataset designed for fine-grained scholarly information extraction in ML research. It features 10 detailed entity types and 18 relation types annotated across the full-text of scientific articles, comprising more than 62K annotated entities and 35K relations. Both data model and dataset shed light on the interdependencies between key concepts involved in machine learning-related research and facilitate research into information extraction methods that can provide a more structured view on machine learning-related artefacts and their relations at scale.

Our experiments show that fine-tuned models trained on this dataset significantly outperform prompting methods using LLMs, highlighting the current limitations of unsupervised approaches in domain-specific IE tasks. The significant performance gap supports our claim that curated datasets like GSAP-ERE are essential to advance reliable extraction of research metadata, enabling downstream applications such as knowledge graph construction or the understanding of reproducibility of ML-based research at scale.

5.8 Limitations

With respect to the coverage of the dataset, we include only publications from (applied) machine learning to simplify the annotation task and focus the required expertise for annotation.
We emphasize the complexity of RE annotation, and will transfer the data curation lessons when we extend the corpus with more diverse fields. In addition, our corpus covers only sentence-level annotations. Although our sentence-level ERE dataset is already challenging due to the multiplicity of entity mentions and relations, future work on a document-level corpus would help advancing the task further. Note that document-level relation extraction also needs document-level coreference resolution, and the coreference clusters may include unnamed generic mentions, which requires an additional annotation step.

Ethical Statement

All documents in this dataset come from public arXiv submissions. These documents are already publicly available under the terms set by arXiv. We removed personal data such as author email addresses and kept only the content needed for research on scientific text processing.

Human annotators were informed about the goals of the project and agreed to take part. No sensitive personal data was collected from annotators. The annotation process followed clear guidelines, and we monitored annotation quality throughout the project.

The model we present can extract entities and relations from scientific text. It is not meant for use on private or sensitive documents. Users should apply it only to text they are allowed to process.

We release the dataset and model under licenses that support research use. We provide documentation to help others understand the limits of the data, possible errors, and the expected behavior of the model.

Acknowledgments

We thank the anonymous reviewers for their constructive feedback. This work has been partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of the Projects BERD@NFDI (grant number 460037581), NFDI4DS (grant number 460234259), as well as Unknown Data (grant number 460676019) and ConnOSS (grant number 561044496).

6

Schema-Preserving Generalization Across Scientific ERE Benchmarks

Publication Information

Title: Generalizing Without Collapsing: Schema-Preserving Transfer Across Scientific ERE Benchmarks

Authors: Wolfgang Otto, Stefan Dietze

Status: Submitted to EACL 2027 (Archival, Peer-Reviewed)

Reuse: This chapter reproduces the text of the publication listed above in thesis layout.

Context

This publication addresses a problem that becomes visible once several scientific ERE benchmarks for ML-related literature are available: model results are difficult to compare across datasets because the datasets differ in entity schemas, relation schemas, corpus distributions, annotation density, and annotation criteria. As a result, high scores on one benchmark do not show whether a model has learned reusable extraction behavior or whether it has adapted to dataset-specific annotation patterns. This chapter therefore treats cross-dataset comparability and generalization as a thesis-level concern, rather than as a technical detail of evaluation. The work compares and aligns SciER, SciNLP, and GSAP-ERE as the current full-text scientific ERE resources for ML-related publications. It first constructs a unified evaluation setting that maps the datasets into a shared label space and enables controlled cross-dataset tests. It then studies whether models trained on one dataset transfer to the others under this shared schema. Beyond this alignment-based view, the chapter evaluates a dataset-conditioned multi-head model that shares representations across datasets while preserving each dataset's original prediction space. The results show that schema unification is useful for comparison, but that relation extraction transfer benefits from preserving dataset-specific schemas instead of collapsing them into one reduced schema. Within the thesis, this chapter connects the earlier entity and relation extraction resources to a broader question: which extraction signals remain stable across scientific ERE datasets, and which signals depend on a specific annotation design. It shifts the thesis line from isolated extraction performance toward robustness and schema-preserving transfer across resources. The chapter also informs later application-oriented work by clarifying which extracted signals can be reused across mixed sources and where dataset-specific interpretation remains necessary.

Abstract

Scientific entity and relation extraction (ERE) benchmarks do not show whether models learn general extraction behavior or adapt to dataset-specific annotation patterns. We study this problem across SciER, SciNLP, and GSAP-ERE, three recent full-text scientific ERE datasets with overlapping targets but different corpus distributions, schemas, and annotation practices. We introduce an alignment pipeline for controlled cross-dataset evaluation and a dataset-conditioned multi-head ERE model that shares representations while preserving each dataset's original prediction space. Our experiments show that schema unification enables direct comparison but is not sufficient for robust transfer, especially for relation extraction, where similarly named labels often encode different annotation semantics. In contrast, schema-preserving multi-head training improves performance on two of the three datasets, setting a new state of the art for relation extraction on SciER and producing large gains on SciNLP. These results show that generalizable scientific ERE requires more than merging datasets: models benefit from shared training signals when dataset-specific schemas are preserved rather than collapsed. We release the unified benchmark, alignment pipeline, model outputs, code, and Dataset Explorer web app to support controlled cross-dataset evaluation.

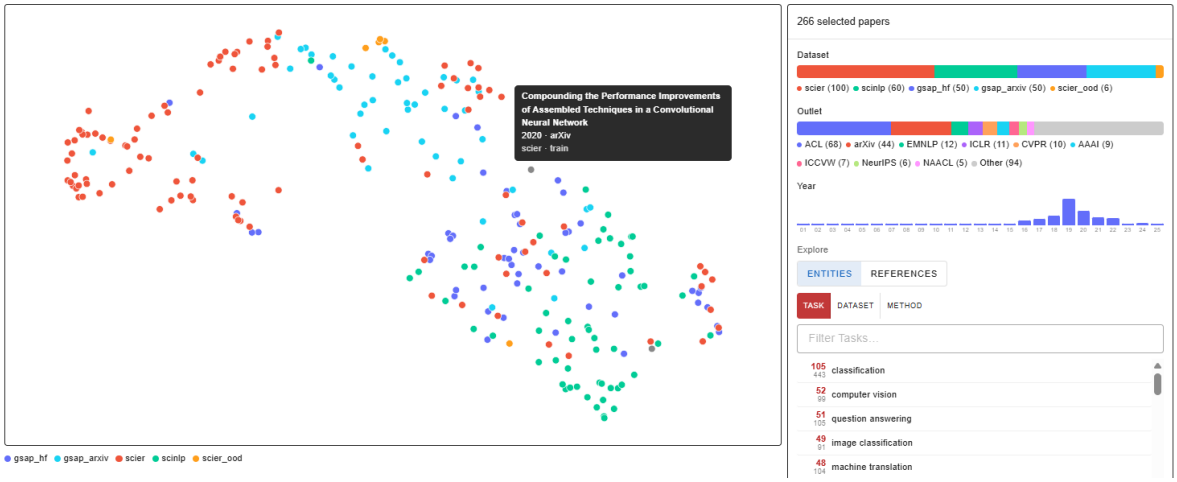


Figure 6.1: Joint corpus-level view of the three scientific ERE datasets in the Dataset Explorer web app. Papers are projected from SPECTER2 scientific-document embeddings (A. Singh et al., 2023) and colored by dataset (i.e. SciNLP, SciER, or GSAP-ERE). The side panel summarizes dataset, outlet, year, and frequent task mentions for the selected papers, showing both overlap and distributional differences between the datasets.

6.1 Introduction

Machine learning and artificial intelligence research is expanding rapidly, making it increasingly difficult to track state-of-the-art methods, benchmark datasets, evaluation practices, and resource dependencies. This problem is especially visible in NLP, where publication volume and community size continue to grow (Abdalla et al., 2023). Understanding this literature requires more than keyword search, because models, datasets, tasks, metrics, and evaluation results are often described informally and across paper full texts rather than in a single standardized format. This is especially important in ML

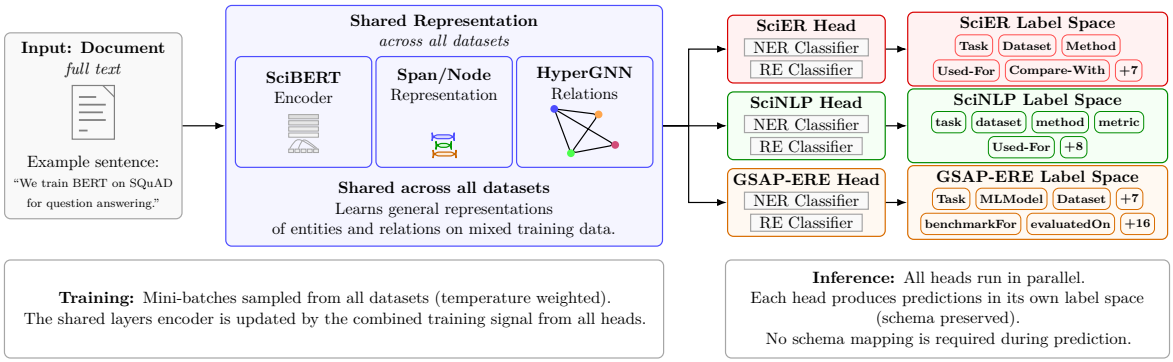


Figure 6.2: *Dataset-conditioned multi-head multi-task architecture.* The encoder, span representations, and HyperGNN are shared across datasets, while each dataset keeps separate NER and RE prediction heads for its original schema.

and AI, where experimental settings, resource dependencies, and evaluation evidence are distributed across different sections of a paper. Incomplete reporting of training data, evaluation data, model dependencies, and metrics also contributes to reproducibility challenges in ML research (Pineau et al., 2021). Scientific ERE addresses this problem by extracting research concepts and their relations from scholarly text. Such structured representations support knowledge graph construction, benchmark tracking, literature review automation, model comparison, scientific question answering, and large-scale analysis of scientific claims, resource usage, and methodological dependencies (Luan et al., 2018; Q. Zhang et al., 2024; Duan et al., 2025; Otto et al., 2026).

Until recently, no benchmark provided manually annotated full-text entity and relation extraction for ML-related scientific publications. SciER, SciNLP, and GSAP-ERE fill this gap and, to our knowledge, are currently the only such resources (Q. Zhang et al., 2024; Duan et al., 2025; Otto et al., 2026). They cover complete papers at the annotation level, while entity and relation decisions remain sentence-level. Q. Zhang et al. (2024) and Otto et al. (2026) also show that general-purpose LLM in-context learning does not solve scientific ERE for ML-related entities: supervised models remain stronger, especially for relation extraction.

The availability of SciER, SciNLP, and GSAP-ERE makes it possible to move beyond isolated benchmark evaluation and ask whether scientific ERE models generalize across annotation settings. This is not guaranteed by high scores on individual datasets alone. Although the datasets target overlapping ML-related concepts, they sample different parts of the ML literature and differ in schema design, span conventions, null-sentence coverage, and relation annotation criteria. Models may therefore learn dataset-specific cues instead of reusable extraction behavior. This concern is related to shortcut learning and recent findings on dataset artifacts in NLP benchmarks (Geirhos et al., 2020; Feger et al., 2025; Štefánik et al., 2025). Since these datasets are costly to create and currently define the benchmark landscape for ML-related scientific ERE, their differences and transfer behavior need to be understood before benchmark scores can be compared or used to assess cross-dataset generalization.

In this work, we study benchmark generalizability in full-text scientific ERE by comparing, aligning, and jointly evaluating recent ML-related benchmark corpora: SciER, SciNLP, and GSAP-ERE. We analyze corpus and annotation differences, test cross-dataset transfer under a shared schema, and investigate joint training strategies. Our

results show that schema unification enables direct comparison but is not sufficient for robust relation extraction, because similarly named relations can encode different annotation semantics. We further show that a dataset-conditioned multi-head setup (Figure 6.2) shares representations across datasets while preserving their original label spaces. This schema-preserving strategy improves performance on SciER and SciNLP, especially for relation extraction, but does not improve performance on GSAP-ERE. The resulting transfer pattern is asymmetric: smaller or coarser benchmarks benefit from shared representations learned with GSAP-ERE, whereas the largest and most fine-grained benchmark gains little from adding less detailed sources.
This paper makes the following contributions:

1. We systematically compare and align all current ML-related scientific ERE benchmark corpora, SciER, SciNLP, and GSAP-ERE, at corpus and schema level, deriving a unified label space for controlled cross-dataset evaluation.
2. We introduce a dataset-conditioned multi-head multi-task approach that shares representations while preserving each dataset's original label space, predicts all three original schemas, and improves over single-dataset training on two datasets, including state-of-the-art relation extraction on SciER.
3. We conduct a reproducibility and cross-dataset generalization study under a shared implementation and evaluation setup, reproducing most HGERE baselines closely, identifying a substantial SciNLP relation-extraction mismatch that motivated corrected reference results, and showing that NER transfers more easily than RE under the unified schema.
4. We release the unified benchmark, schema alignment pipeline, model outputs, code, and Dataset Explorer web app[1] for inspecting corpus composition, data models, paper-level predictions, and cross-dataset comparisons.[2]

6.2 Related Work

Scientific ERE aims to identify research concepts such as datasets, methods, models, tasks, and metrics, and to link them through relations that capture usage, evaluation, comparison, and conceptual dependencies. This structured information supports search, survey work, benchmark monitoring, knowledge graph construction, and scientific question answering over ML/NLP literature (Q. Zhang et al., 2024; Duan et al., 2025; Otto et al., 2026).
Early scientific IE benchmarks established common entity and relation extraction settings, but typically focused on abstracts or selected paragraphs rather than complete papers (Augenstein et al., 2017; Gábor et al., 2018; Luan et al., 2018). This limits coverage of experimental details, recurring mentions, section-specific evidence, and relations that are only expressed in the body of a paper. More recent full-text resources therefore address a broader annotation setting, but also introduce new challenges such as long inputs, dense technical terminology, aliases, informal mentions, and dataset-specific annotation choices (Jain et al., 2020; Otto et al., 2023).
SciER, SciNLP, and GSAP-ERE represent this recent shift toward manually annotated,

[1] Dataset Explorer: https://huggingface.co/spaces/paper938487/companion-app

[2] We publish our code, including forks, under the MIT license: https://github.com/ottowg/generalizing-sciere-benchmarks, https://github.com/ottowg/gsapere

	SciER	SciNLP	GSAP-ERE
# Publications	106	60	100
# Split	80/10/10/6	51/6/3	80/10/10
# Entity Types	3	4	10
# Relation Types	9	9	18
# Entities	24,518	6,409	62,619
# Relations	12,083	1,648	35,302
# Entities in Relations	17,148 (70%)	2550 (40%)	46,680 (75%)
# Entity Sentences	7,722	3,688	20,020
# Null Sentences	2	9,526	6,185
# Entities/Pub.	231.3	106.8	626.2
# Relations/Pub.	114.0	27.5	353.0
# Sentences/Pub.	72.9	220.2	262.1

Table 6.1: Comparison of the examined scientific Entity and Relation Extraction (ERE) datasets for scientific text SciER, SciNLP, and GSAP-ERE. Note that GSAP-ERE is the biggest dataset in terms of annotated entities and relations and SciER is missing checked *null Sentences* without entity mentions as negative samples even in the dev and test set. The figures for SciNLP are corrected counts based on the dataset published on github: https://github.com/AKADDC/SciNLP and the corrected arXiv version Duan et al., 2026 the EMNLP version Duan et al., 2025 stated 7,072 entities and 1,826 relationships.

full-text scientific ERE for ML-related publications (Q. Zhang et al., 2024; Duan et al., 2025; Otto et al., 2026). Although they target overlapping concepts, they differ in corpus selection, entity inventories, relation definitions, annotation density, null-sentence coverage, and schema granularity. These differences make reported scores difficult to compare directly and motivate the dataset analysis and alignment in Section 6.3 and Appendix 6.8.1.

Most recent scientific ERE systems build on pretrained scientific encoders such as SciBERT and model extraction as span-based entity typing plus relation classification over entity pairs (Beltagy et al., 2019). Pipeline models such as PURE and marker-based models such as PL-Marker are common baselines, while joint approaches such as HGERE combine entity and relation decisions in a shared model (Zhong and Chen, 2021; D. Ye et al., 2022; Yan et al., 2023). However, the relative performance of these approaches depends on schema granularity, negative-instance coverage, and evaluation strictness, which reinforces the need for controlled cross-dataset protocols.

6.3 Dataset Analysis

6.3.1 Benchmark Differences

Paper distributions. The datasets are not sampled from a single common paper distribution. SciER covers broad AI topics, including NLP, computer vision, machine learning, and AI4Science (Q. Zhang et al., 2024). SciNLP focuses on ACL/NLP papers from 2001 to 2024 (Duan et al., 2025), while GSAP-ERE samples machine learning

and applied machine learning papers from arXiv and Hugging Face model README references (Otto et al., 2026). Figure 6.1 shows that the datasets overlap in parts of the ML literature, but form separate regions. Thus, cross-dataset evaluation is affected by corpus selection and reporting style, not only by schema design.

Schema design and annotation criteria. Table 6.1 summarizes the main schema differences. SciER uses a compact schema with three entity types and nine relation types focused on datasets, methods, and tasks (Q. Zhang et al., 2024). SciNLP extends this core schema with a metric type and relations tailored to evaluation and model usage (Duan et al., 2025). GSAP-ERE is the most fine-grained resource, with ten entity types and eighteen relation types covering model, generic-mention, data, provenance, reference, and peer-relation information (Otto et al., 2026). Thus, similarly named labels are not necessarily interchangeable. For example, method- or model-like concepts differ in whether model architectures, generic model mentions, and methods are separated or collapsed. Relation labels can also differ in directionality, domain and range constraints, or criteria for when a relation should be annotated. As a result, models may learn dataset-specific annotation conventions rather than general scientific ERE behavior.

Supervision density and negative evidence. The datasets also differ strongly in annotation density and negative evidence. Table 6.1 shows that GSAP-ERE is both larger overall and much denser per document. Normalized by publication, it provides about 5.9 times as many entity annotations and 12.8 times as many relation annotations as SciNLP, changing the amount of local supervision available for learning. SciNLP also has an uneven temporal profile, with sparse early ACL coverage and stronger concentration in recent years, which may affect transfer across changing terminology and reporting practices. SciER contains almost no checked null sentences, whereas SciNLP and GSAP-ERE preserve many sentences without target annotations. These null sentences account for 72% of SciNLP and 24% of GSAP-ERE, changing the negative-instance distribution for both entity and relation extraction. The effect is especially relevant for relation extraction, where many candidate entity pairs express no target relation.

6.3.2 What Can Be Aligned and What Should Be Preserved

The datasets share a small conceptual core: task, dataset, and method- or model-like entities, plus relations for usage, evaluation, comparison, and hierarchy. This shared core supports a unified schema for controlled cross-dataset evaluation. However, alignment also collapses source-specific distinctions, such as GSAP-ERE's model, architecture, generic, data-source, reference, and URL entities, or SciNLP's separate metric type. Relation alignment is even more difficult, because labels with similar names can differ in semantics, directionality, and annotation criteria. Our experiments therefore use complementary strategies: unified-schema evaluation tests what can be compared after alignment, while dataset-conditioned multi-head training tests what can transfer when each dataset's original prediction space is preserved.

6.4 Experimental Setup

6.4.1 Task definition

We define ERE as named entity recognition (NER) over labeled spans and relation extraction (RE) over pairs of entity mentions. Let $\mathcal{L}_E$ denote the set of entity labels and $\mathcal{L}_R$ the set of relation labels. Given a document $D = \{p_1, \ldots, p_n\}$ represented by sentences p_i, the NER subtask predicts labeled entity spans $m = (b, d, \ell)$ for each sentence p_i, where b and d denote the begin and end token indices of the entity span and $\ell \in \mathcal{L}_E$ denotes its entity label. The RE subtask predicts relation triples (m_i, r, m_j), where m_i and m_j are entity mentions and $r \in \mathcal{L}_R$. Although the datasets cover full papers, prediction follows the source annotations and is sentence-level: entities are predicted per sentence and relations only between sentence-level mentions.

6.4.2 Data splits and preprocessing

Datasets and splits. We use SciER, SciNLP, and GSAP-ERE in their original annotation schemas and label sets. We keep the released train, development, and test splits for all single-dataset experiments. Notably, the SciNLP test set is small (3 documents), but we retain it unchanged to ensure direct comparability with prior work. For cross-dataset evaluation, we map predictions and gold annotations of each dataset into the unified schema using the unification pipeline (Appendix 6.8.1). This yields unified versions $\text{SciER}_{\text{uni}}$, $\text{SciNLP}_{\text{uni}}$, and $\text{GSAP-ERE}_{\text{uni}}$, which are evaluated separately under a shared label space. To study joint training using the unified labels, we construct $All_{unified}$ by merging the unified train, development, and test splits of the three datasets into corresponding combined splits. Hence, models trained on $All_{unified}$ can only be evaluated on the unified test sets.

Preprocessing. We preserve each dataset's original tokenization for comparability with the released versions. Transformer subtokens are obtained by re-tokenizing the provided token sequences.

6.4.3 Metrics.

Following Otto et al. (2026), we report exact and partial-span variants for NER and RE. **NER** requires exact span and label match, while **NER≈** allows overlapping spans. For relations, **RE** requires exact subject and object spans plus the relation label, while **RE+** additionally requires correct entity labels. The partial variants **RE≈** and **RE+≈**

Setup	Epochs	Learning rate	Batch size
SciER	20	5e-5/5e-5	18
SciNLP	40	1e-5/1e-4	32
GSAP-ERE	8	1e-5/1e-5	21
$All_{unified}$	10	1e-5/1e-5	21
MultiSciERE	8	1e-5/1e-5	21

Table 6.2: Final hyperparameters used for training.

Dataset	Task	Repr.	Paper	Δ
GSAP-ERE	NER	82.8 ± 0.7	80.6 ± 0.3	+2.2
	RE	55.7 ± 0.4	54.0	+1.7
	RE+	47.9 ± 0.5	46.9 ± 0.5	+1.0
SciER	NER	87.2 ± 0.4	86.85	+0.4
	RE	64.9 ± 1.2	62.32	+2.6
	RE+	62.6 ± 1.4	61.10	+1.5
SciNLP	NER	80.3 ± 1.2	79.53	+0.8
	RE	45.7 ± 2.9	49.28	−3.6
	RE+	45.4 ± 2.7	47.64	−2.2

Table 6.3: Results of our reproduction of the dataset paper baselines (micro-F1; %). Reported values include standard deviations where available. "+" indicates the strict RE metric (RE+), where entity types are also required to match.

replace exact entity-span matching with overlap. Here, ≈ denotes partial span matching and + denotes the stricter variant that additionally requires correct entity typing. For symmetric relations, we normalize and de-duplicate gold and predicted pairs to a left-to-right order before evaluation. Unless stated otherwise, all reported results are micro-averaged F1 scores. All experiments use five seeds, reporting mean and standard deviation rounded to one decimal. Significance tests use unrounded seed-level scores.

6.4.4 Model and Training

All experiments use HGERE with a SciBERT backbone (Yan et al., 2023; Beltagy et al., 2019). HGERE is a strong joint ERE model representing the state of the art for SciER, SciNLP, and GSAP-ERE (Q. Zhang et al., 2024; Duan et al., 2025; Otto et al., 2026). This makes it a suitable reference model for studying schema alignment and cross-dataset transfer. For multi-head training, we share the SciBERT encoder, span representations, and HyperGNN relation representations across datasets. Only the final NER and RE classifiers are dataset-specific. During training, each batch comes from one dataset and uses the corresponding head; datasets are interleaved with temperature-weighted sampling (Raffel et al., 2020; Y. Liu et al., 2020). At inference time, the dataset identity selects the relevant head, while the shared model can also run all heads in parallel. This design follows prior work on multi-task and multi-head learning for handling label variation across datasets (Pham et al., 2023; Straková and Straka, 2025).

We forked the HGERE implementation of Yan et al. (2023). Training all compared five-seed models took approximately 48h on one NVIDIA H100 GPU. We tune epochs, learning rate, and batch size per setup; final values are shown in Table 6.2, and full configurations are available in our repository.[3]

6.4.5 Experimental settings

We study four complementary setups to analyze reproducibility, cross-dataset comparability, and joint learning across heterogeneous annotation schemas.

(i) Single-dataset training. We train and evaluate models on each dataset separately using their original annotation schemas and splits. This setup serves as a reproduction baseline and establishes a reference point for comparison with the results reported in the original dataset papers.

(ii) Cross-dataset evaluation and unified training. We train a single model on the merged $All_{unified}$ dataset under the unified label schema and test on each of the datasets separately. This setup tests whether combining data across datasets improves performance through increased training data and domain diversity.
We evaluate the models from (i) under the unified schema by mapping their predictions into the unified label space. We also map the single-dataset models from setup (i) into the unified schema and evaluate them across datasets. This tests whether models trained on one source dataset, or on $All_{unified}$, transfer to the others without retraining.

(iii) Dataset-conditioned multi-head training. We train a joint model on all datasets while preserving their original label spaces via dataset-specific prediction heads. All representation layers are shared, while final predictions remain dataset-specific. This setup tests whether shared representations can be learned across datasets without enforcing a unified schema, and whether this improves performance compared to (i) and (ii).

6.5 Results

6.5.1 Reproducibility and Unified Evaluation

We first re-run HGERE on each dataset with a common implementation and evaluation setup. We compare the resulting original-schema scores against the best HGERE scores reported in the dataset papers and additionally evaluate the same models after mapping gold annotations and predictions to the unified schema (Q. Zhang et al., 2024; Duan et al., 2026; Otto et al., 2026). Since the original papers do not report seed-level scores, we do not test significance against the published baselines. Table 6.3 shows that our reproduction closely matches the published HGERE baselines. For GSAP-ERE and SciER, all reproduced scores are slightly above the reported values. SciNLP is the only exception: NER is slightly higher than in the corrected arXiv version, but RE and RE+ remain lower. This comparison uses the corrected arXiv version of SciNLP, which

[3] `configs/<dataset_name>/train/hgere/best.yaml` in `https://github.com/ottowg/gsapere`.

Test set	Trained on	NER	NER≈	RE	RE+
SciNLP$_{\text{uni.}}$	SciER	87.2 ± 0.4	91.5 ± 0.2	**65.4 ± 1.2**	**63.1 ± 1.4**
	GSAP	79.2 ± 2.4 *(Δ -8.0)*	86.5 ± 1.3 *(Δ -5.0)*	34.3 ± 3.1 *(Δ -31.1)*	33.9 ± 3.1 *(Δ -29.2)*
	SciNLP	71.9 ± 0.5 *(Δ -15.3)*	83.9 ± 0.7 *(Δ -7.6)*	32.4 ± 1.4 *(Δ -33.0)*	31.7 ± 1.3 *(Δ -31.4)*
	All$_{\text{uni}}$	**89.4 ± 0.4** ***(Δ +2.2)***	**93.0 ± 0.3** ***(Δ +1.5)***	44.2 ± 2.5 *(Δ -21.2)*	43.1 ± 2.3 *(Δ -20.0)*
SciNLP$_{\text{uni.}}$	SciNLP	**80.2 ± 1.4**	84.5 ± 1.3	**44.4 ± 2.7**	**44.1 ± 2.6**
	GSAP	64.9 ± 2.1 *(Δ -15.3)*	69.1 ± 1.9 *(Δ -15.4)*	31.1 ± 2.8 *(Δ -13.3)*	29.9 ± 2.5 *(Δ -14.2)*
	SciER	73.3 ± 1.3 *(Δ -6.9)*	77.4 ± 1.0 *(Δ -7.1)*	40.7 ± 2.0 *(Δ -3.7)*	38.4 ± 2.0 *(Δ -5.7)*
	All$_{\text{uni}}$	80.0 ± 0.6 (Δ -0.2)	**85.0 ± 0.6** (Δ **+0.5**)	35.6 ± 2.7 *(Δ -8.8)*	34.7 ± 3.0 *(Δ -9.4)*
GSAP$_{\text{uni.}}$	GSAP	**90.9 ± 1.0**	**93.7 ± 0.5**	**59.6 ± 0.4**	**58.8 ± 0.4**
	SciER	59.8 ± 1.0 *(Δ -31.1)*	68.6 ± 0.9 *(Δ -25.1)*	29.7 ± 0.7 *(Δ -29.9)*	28.7 ± 0.9 *(Δ -30.1)*
	SciNLP	52.1 ± 0.7 *(Δ -38.8)*	60.9 ± 0.4 *(Δ -32.8)*	17.6 ± 0.7 *(Δ -42.0)*	16.9 ± 0.9 *(Δ -41.9)*
	All$_{\text{uni.}}$	69.7 ± 0.4 *(Δ -21.2)*	75.8 ± 0.3 *(Δ -17.9)*	23.8 ± 1.7 *(Δ -35.8)*	22.9 ± 1.7 *(Δ -35.9)*

Table 6.4: Cross-dataset and unified training results on the unified labelset (micro-F1; %). Deltas are computed relative to the single-dataset baseline for each test set. Best scores per test set and metric are shown in bold.

reports much lower HGERE scores than the earlier ACL version (Duan et al., 2025; Duan et al., 2026). This confirms the general reproducibility of the baselines, while indicating that SciNLP relation extraction is less stable in our setup. A likely reason is the small SciNLP split size, which is also reflected in the higher variance of its NER and RE scores.

Unified evaluation changes task difficulty in a dataset-dependent way. The strongest effect appears for GSAP-ERE, where reducing the fine-grained original schema to the shared schema increases NER, RE, and RE+. SciER changes only marginally, while SciNLP becomes slightly weaker after mapping. Thus, schema alignment enables controlled comparison, but it does not affect all datasets equally.

We next use the unified schema to evaluate how well models generalize across datasets, as shown in Table 6.4. It compares single-dataset models mapped to the unified schema with a model trained on the merged unified training data. This setup tests whether schema unification is sufficient to support transfer across datasets despite differences in their underlying document distributions and annotation decisions. If transfer performance remains below the same-dataset baselines, this may reflect not only remaining differences in labeling behavior, but also dataset-specific distributions and evidence patterns. The results show that unification aligns broad entity concepts

Test set	Trained on	NER	NER≈	RE	RE+
SciER$_{\text{orig.}}$	SciER	87.2 ± 0.4	91.5 ± 0.2	64.9 ± 1.2	62.6 ± 1.4
	SciER+SciNLP	90.2 ± 0.3 *(Δ +3.0)*	92.9 ± 0.2 *(Δ +1.4)*	68.0 ± 0.7 *(Δ +3.1)*	66.7 ± 0.9 *(Δ +4.1)*
	SciER+GSAP	**91.2 ± 0.3** ***(Δ +4.0)***	**93.7 ± 0.2** ***(Δ +2.2)***	68.4 ± 0.7 *(Δ +3.5)*	67.4 ± 0.6 *(Δ +4.8)*
	All$_{\text{orig.}}$	90.9 ± 0.2 *(Δ +3.8)*	93.4 ± 0.2 *(Δ +2.0)*	**69.1 ± 0.7** ***(Δ +4.1)***	**68.1 ± 0.7** ***(Δ +5.5)***
SciNLP$_{\text{orig.}}$	SciNLP	80.3 ± 1.2	84.1 ± 1.1	45.7 ± 2.9	45.4 ± 2.7
	SciNLP+GSAP	**86.7 ± 1.5** ***(Δ +6.4)***	**89.4 ± 0.6** ***(Δ +5.2)***	52.6 ± 2.2 *(Δ +7.0)*	51.1 ± 2.6 *(Δ +5.7)*
	SciNLP+SciER	83.4 ± 1.8 *(Δ +3.1)*	85.7 ± 1.9 *(Δ +1.6)*	44.1 ± 2.6 (Δ -1.5)	43.7 ± 2.6 (Δ -1.7)
	All$_{\text{orig.}}$	85.6 ± 1.4 *(Δ +5.3)*	88.5 ± 1.1 *(Δ +4.3)*	**54.0 ± 1.6** ***(Δ +8.3)***	**52.4 ± 1.4** ***(Δ +7.0)***
GSAP$_{\text{orig.}}$	GSAP	82.8 ± 0.7	86.3 ± 0.6	55.7 ± 0.4	47.9 ± 0.5
	GSAP+SciNLP	82.1 ± 0.5 (Δ -0.7)	85.6 ± 0.3 (Δ -0.6)	53.7 ± 0.8 *(Δ -2.0)*	46.2 ± 0.7 *(Δ -1.6)*
	GSAP+SciER	**83.0 ± 0.5** (Δ **+0.2**)	**86.3 ± 0.5** (Δ +0.0)	**55.9 ± 0.5** (Δ **+0.2**)	**48.2 ± 0.9** (Δ **+0.3**)
	All$_{\text{orig.}}$	80.7 ± 0.5 *(Δ -2.1)*	84.3 ± 0.6 *(Δ -1.9)*	53.7 ± 0.7 *(Δ -1.9)*	44.9 ± 0.7 *(Δ -3.0)*

Table 6.5: MultiSciERE results on entity and relation extraction grouped by test set (micro-F1; %). This model is trained on 1-3 datasets and makes use of each of the different label sets in a multi-head multi-task learning architecture build upon a HGERE Yan et al., 2023 model. Best scores within each test set and metric are shown in bold. Significant deltas are shown in italics. NER≈ denotes partial span matching.

more effectively than relation-level annotation behavior.

For NER, cross-dataset transfer remains below the same-dataset baselines under exact matching, but partial-span scores are often much closer. For example, transfer from SciNLP$_{\text{uni}}$ to SciER$_{\text{uni}}$ increases from 71.9 ± 0.5 NER to 83.9 ± 0.7 NER≈, a gap of 12.0 F1 points. This suggests that many NER transfer errors concern span boundaries rather than the underlying entity concept. The unified model is generally closer to the same-dataset NER baselines than most single-source transfer models, especially on SciER$_{\text{uni}}$ and SciNLP$_{\text{uni}}$, but this stabilizing effect does not carry over to RE. For relations, cross-dataset RE and RE+ scores remain far below the same-dataset baselines for most train–test pairs. These results suggest that unified relation labels alone are not sufficient to make relation extraction behavior directly transferable across datasets. The observed performance differences may reflect differences in relation semantics, annotation practices, document distributions, and evidence patterns. Thus, unification provides a useful comparison layer, but it does not remove dataset-specific effects. This motivates the multi-head setting in the next section, where the model shares representations across datasets while keeping dataset-specific prediction heads.

6.5.2 Multi-Head Transfer Evaluation

Table 6.5 evaluates transfer when datasets keep their original label spaces through dataset-specific prediction heads. This setup tests whether shared representations can improve transfer without forcing different annotation schemas into a single unified label space. The results show that multi-head training is more effective than unified merging for SciER and SciNLP, but not for GSAP-ERE.

For SciER, multi-head training improves all metrics over the single-dataset baseline, with the all-dataset model reaching the best relation scores. RE increases by +4.1 F1 and RE+ by +5.5 F1. SciNLP benefits even more from transfer, with the largest relation gains in Table 6.5: +8.3 RE and +7.0 RE+. NER also improves on both datasets, with the best scores obtained by models that include GSAP-ERE.

GSAP-ERE shows the opposite pattern. The single-dataset GSAP-ERE model remains strongest for most metrics, while the all-dataset model is lower across all four metrics. For example, RE+ decreases from 47.9 ± 0.5 to 44.9 ± 0.7 when all datasets are used. The pairwise GSAP+SciER model is effectively neutral, whereas adding SciNLP lowers performance. Thus, GSAP-ERE receives little benefit from the smaller or less compatible sources and can be harmed when all sources are combined.

Overall, multi-head transfer is not always beneficial. It improves SciER and SciNLP, especially for relation extraction, while GSAP-ERE performs best when trained only on its own training data. For SciER, the all-dataset multi-head model establishes a new state of the art under the original benchmark setting, improving RE to 69.1 ± 0.7 and RE+ to 68.1 ± 0.7. For SciNLP, the model substantially improves over the corrected HGERE reference, but we avoid a state-of-the-art claim because the corrected arXiv version does not revise the previously reported PL-Marker relation scores.

These results show that preserving original label spaces is important for transfer. Unlike unified training, where relation transfer remains weak, the multi-head model shares representations while keeping dataset-specific output spaces. The gains also indicate an asymmetric transfer pattern: SciER and SciNLP benefit from additional training signals, especially when GSAP-ERE is included, whereas GSAP-ERE itself gains little or decreases. This suggests that fine-grained resources can support prediction heads with less detailed schemas, but joint training does not necessarily improve the most fine-grained target itself.

6.6 Discussion

Our results show that full-text scientific ERE benchmarks share useful extraction signals, but these signals are only partly captured by a single unified schema. Unified mapping enables controlled comparison across SciER, SciNLP, and GSAP-ERE, but it also removes distinctions needed for relation extraction, especially when similar label names encode different evidence rules, argument choices, or annotation scopes across datasets. Thus, schema unification is useful as a joint evaluation layer, but loses information when used as a training schema. The multi-head setting provides a more effective alternative for joint training. By sharing representations while preserving dataset-specific prediction heads, the model can transfer extraction behavior without forcing all labels into one reduced output space. This benefits SciER and SciNLP, especially for relation extraction, while GSAP-ERE does not improve under full multi-dataset

training. This asymmetric pattern suggests that larger and more fine-grained resources can provide useful auxiliary signal for datasets with less detailed schemas, while the reverse direction provides less benefit. These findings suggest that future benchmark design should distinguish conceptual overlap from annotation equivalence. Rather than treating schema variation as noise, scientific ERE evaluations should support aligned but schema-preserving comparison. Because the multi-head model preserves all original output spaces, it also enables future work on combining predictions across schemas instead of forcing all predictions into a single reduced label space.

6.7 Conclusion

We introduced a unified resource and evaluation framework for transfer across SciER, SciNLP, and GSAP-ERE. Unified schema mapping enables direct comparison, but does not provide robust relation extraction transfer. Dataset-conditioned multi-head training instead shares representations while preserving original label spaces. This improves SciER and SciNLP, especially for relation extraction, while GSAP-ERE remains best served by its own fine-grained training data. Overall, generalizable scientific ERE requires shared training signals without collapsing dataset-specific schema distinctions.

Limitations

This study has several limitations. First, although all three source datasets are based on full scientific papers, their extraction targets remain sentence-level. Here, "full-text" refers to annotation coverage across complete papers, not to document-level relation targets. Our results should therefore be interpreted as transfer results for full-text-derived sentence-level ERE, not as evidence about cross-sentence relations, document-level coreference, n-ary scientific events, tables, figures, or citation-graph signals.

Second, the unified schema necessarily simplifies the original datasets. Schema alignment enables cross-dataset comparison, but it also drops labels, merges relation types, reduces entity granularity, and changes some directionality choices. This is especially relevant for GSAP-ERE, whose original schema contains more fine-grained entity and relation distinctions. Moreover, similar label names do not guarantee equivalent annotation semantics, since relations can differ in evidence rules, linkable entity pairs, and treatment of implicit relations.

Third, our evaluation and modeling choices cover only part of the scientific ERE problem. Exact and partial span metrics expose different aspects of boundary variation, typed and untyped relation metrics differ in how they handle propagated NER errors, and micro-averaged F1 can hide failures on rare labels. All main experiments use HGERE with SciBERT, so the observed transfer patterns should be interpreted as evidence for one strong joint ERE architecture rather than as model-independent results.

Finally, the datasets differ in domain, time span, venue coverage, split size, null-sentence coverage, tokenization, and PDF-to-text conversion. We preserve the released splits and tokenization to remain comparable to the source datasets, but this leaves preprocessing differences as a possible source of variation. The reproducibility study

is also limited by the information available in the original releases: we reproduce the reported HGERE baselines closely for most settings, but SciNLP relation extraction is not fully reproduced under our setup. These factors mainly limit how strongly individual score differences should be interpreted.

Ethical and Artifact Statement

This work reuses existing scientific ERE datasets: SciER, SciNLP, and GSAP-ERE (Q. Zhang et al., 2024; Duan et al., 2026; Otto et al., 2026). These datasets are derived from publicly available scientific publications and were originally released for research on scientific information extraction. We do not collect new documents, create new annotations, introduce human-subject data, or annotate private or sensitive documents.

Our released artifacts consist of aligned dataset representations, schema mappings, experimental code, model outputs, and an inspection application for research on scientific text processing. They may include paper-level bibliographic metadata needed for provenance and reproducibility, such as paper identifiers, titles, publication years, venues, source dataset information, and split assignments. If author names or affiliations are present in the original public metadata, we treat them only as bibliographic metadata and do not use them for profiling, ranking individuals, or inferring personal attributes.

The released model outputs may contain extraction errors and should be treated as research outputs for analysis and comparison, not as verified factual databases. Users should follow the licenses and terms of the underlying datasets and source documents.

Artifact use and intended use. We use the existing SciER, SciNLP, and GSAP-ERE[4] datasets for scientific entity and relation extraction research, benchmark comparison, schema alignment, and model evaluation. This use is consistent with their intended role as research artifacts for developing and evaluating scientific information extraction methods (Q. Zhang et al., 2024; Duan et al., 2025; Otto et al., 2026). The artifacts created in this work, including the unified benchmark view, schema mappings, model outputs, code, and Dataset Explorer, are intended for research on scientific ERE, benchmark comparability, and cross-dataset generalization. Because our unified resources derive from the original datasets, their use should remain compatible with the access conditions and licenses of the source datasets. In particular, derivatives of data accessed for research purposes should be used only in compatible research contexts and should not be redistributed or applied in ways that violate the original access conditions.

Acknowledgements

We used ChatGPT 5.5 Thinking for reformulation and shortening of author-written text. We used Claude Sonnet 4.6 for coding support. All authors reviewed and edited the resulting text and code, and take responsibility for the final content of the paper.

[4]GSAP-ERE dataset: https://doi.org/10.60914/c4c1d-s0587

6.8 Chapter Appendix

6.8.1 Unified Schema and Dataset Construction

We define a unified schema to make annotations from SciER, SciNLP, and GSAP-ERE comparable in training and evaluation. The unified schema keeps only entity and relation types that can be supported across all three sources, while preserving source provenance for analysis. The unification pipeline is applied per label origin: gold annotations are unified with the mapping rules of the dataset in which they were annotated, while model predictions are unified with the mapping rules of the dataset on which the model was trained. This avoids mixing source-specific annotation assumptions before evaluation. Code, configuration files, and analysis scripts for the pipeline are available in the anonymous repository: `https://github.com/ottowg/generalizing-sciere-benchmarks`.

Unified Entity and Relation Inventory. The unified entity inventory contains three labels: Task, Dataset, and Method. These labels form the shared conceptual core of the three datasets. SciER already uses this compact inventory. SciNLP contributes Task, Dataset, and Model, with Model mapped to Method. GSAP-ERE uses a more detailed inventory, where Method, MLModel, MLModelGeneric, and ModelArchitecture are mapped to Method, and DataSource is mapped to Dataset. Entity labels without stable counterparts in all datasets are removed. This affects DatasetGeneric, ReferenceLink, and URL in GSAP-ERE, and Metric in SciNLP. No SciER entity label is removed.

For relations, we use a shared inventory of seven relation types. The unified labels are oriented toward the GSAP-ERE naming scheme because it provides explicit relation definitions for ML workflows. Relations with matching semantics are renamed, and relations whose direction differs across datasets are inverted where needed. Symmetric relations such as coreference and isComparedTo are stored with a deterministic left-to-right order based on mention position. This removes duplicates with swapped arguments and ensures that symmetric predictions are evaluated consistently.

Some relation types are removed because they depend on entity labels that are not part of the unified schema or because they have no stable equivalent across datasets. Removed relation types include citation, generatedBy, size, and sourcedFrom in GSAP-ERE, and MeasuredBy and evaluatedBy in SciNLP. The resulting schema is therefore a conservative comparison layer, not a replacement for the original dataset schemas.

Pipeline Steps. The unification pipeline consists of eight deterministic steps. First, overlapping or stacked mentions within the same sentence are merged. By default, the larger span is retained. Relations whose arguments collapse into the same mention become self-loops and are removed, and duplicate relations are discarded.

Second, entity labels without a unified equivalent are removed. All relations attached to removed mentions are also removed. This step removes DatasetGeneric, ReferenceLink, and URL from GSAP-ERE, and Metric from SciNLP. No SciER entity label is removed.

Third, the remaining entity labels are mapped to Task, Dataset, and Method. The original label is preserved as metadata to support traceability. For example, GSAP-

Dataset	Ent. orig.	Ent. kept	Ent. %	Rel. orig.	Rel. kept	Rel. %
GSAP-ERE	62,244	31,495	50.6	34,902	10,809	31.0
SciER	23,223	23,223	100.0	11,501	11,492	99.9
SciNLP	6,409	5,432	84.8	1,648	1,366	82.9

Table 6.6: Entity and relation retention after applying the full unification pipeline. Counts are aggregated across train, development, and test splits.

ERE labels such as MLModel, ModelArchitecture, and MLModelGeneric are mapped to Method.

Fourth, relation labels without a unified equivalent are removed. This removes source-specific relation types that cannot be evaluated consistently across all datasets. Examples include provenance-, reference-, and metric-related relations.

Fifth, the remaining relation labels are mapped to the unified relation inventory. Where relation direction differs between a source dataset and the unified schema, subject and object are swapped. Undirected relations are normalized by mention order.

Sixth, we apply GSAP-ERE-specific corrections. This correction is applied only to GSAP-ERE predictions, not to gold annotations. It removes frequent generic MLModelGeneric mentions such as "model", "baseline", or "our system" that are semantically weak and often appear as unmatched cross-dataset predictions. The filter combines a data-driven list of high-frequency unmatched predictions with a static blacklist and simple pattern rules.

Seventh, spans are normalized to reduce systematic boundary differences between datasets. The rules strip recurring prefixes and suffixes that reflect annotation conventions rather than semantic differences. For example, leading determiners such as "the", "a", or "our" may be removed from method mentions, and dataset-specific suffixes such as "dataset", "benchmark", or "corpus" may be removed. Mentions reduced to a single token are not stripped further.

Eighth, a second GSAP-ERE blacklist pass is applied after span normalization. This catches mentions that become generic only after prefix or suffix removal, such as "their model" becoming "model".

The pipeline is controlled through configuration files. These files specify which steps are enabled and define dataset-specific resources for mapping, correction, and reporting. The implementation records step-wise statistics for merged mentions, dropped labels, remapped labels, filtered predictions, span-normalized mentions, and retained annotations. These reports make the transformation from source annotations to the unified schema transparent and reproducible.

Retention after Unification. Table 6.6 summarizes how many entities and relations remain after applying the full pipeline. The retention rates show that unification affects the datasets differently. SciER is almost unchanged because its original schema is already close to the unified schema. SciNLP loses mainly its metric annotations and metric-related relations. GSAP-ERE is affected most strongly because many of its fine-grained entity and relation types are outside the shared schema.

Table 6.7 gives a more detailed entity-level breakdown. For GSAP-ERE, most removed entities are unmapped labels, and additional mentions are removed by the GSAP-ERE-specific correction step. For SciNLP, removed entities correspond mainly

Dataset	Original	Removed		Span-normalized	Kept
		Unmapped	Corrections		
GSAP-ERE	62,244	19,882	2,516	10,362	31,495
SciER	23,223	0	0	4	23,223
SciNLP	6,409	949	0	424	5,432

Table 6.7: Entity retention breakdown across the unification pipeline. Span-normalized mentions are mentions whose surface form is changed by normalization; they are not necessarily removed.

to the unmapped Metric label. For SciER, no entity label is removed, and span normalization changes only a very small number of mentions.

Table 6.8 shows relation retention. The lower relation retention for GSAP-ERE follows mainly from dropped entity labels and relation types outside the unified schema. By contrast, almost all SciER relations are retained. SciNLP loses a smaller but still visible share of relations because metric-related relations are removed.

The dropped entity labels are summarized in Table 6.9. This table makes explicit which source-specific entity types are excluded from the unified schema.

Construction of $\text{All}_{\text{unified}}$. After unification, we construct $\text{All}_{\text{unified}}$ by combining the unified versions of the original train, development, and test splits. We retain the original split assignments and source-dataset provenance. Formally, we define:

$$\begin{aligned} \text{train}_u &= \text{train}_{\text{SciER}} \cup \text{train}_{\text{SciNLP}} \cup \text{train}_{\text{GSAP}}, \\ \text{dev}_u &= \text{dev}_{\text{SciER}} \cup \text{dev}_{\text{SciNLP}} \cup \text{dev}_{\text{GSAP}}, \\ \text{test}_u &= \text{test}_{\text{SciER}} \cup \text{test}_{\text{SciNLP}} \cup \text{test}_{\text{GSAP}}. \end{aligned} \tag{6.1}$$

In the experiments, we report both aggregate results and per-source results, such as evaluation on $\text{SciER}_{\text{unified}}$ or $\text{GSAP-ERE}_{\text{unified}}$. This keeps unified training comparable while still allowing dataset-specific transfer effects to be inspected.

6.8.2 Deriving the Unified Mapping

Entity Mapping Methodology. We derive the unified entity mapping using a model-driven comparison procedure. The goal is to support manual alignment decisions with empirical evidence from cross-dataset predictions.

For each dataset, we train a supervised entity recognition model using the original training split and annotation schema. We then apply each trained model to the development sets of the other datasets. This produces predictions under mismatched schemas and makes systematic differences between entity inventories visible.

For every ordered pair of datasets, we construct confusion matrices that compare predicted entity labels to gold entity labels. These matrices show how often a label from one dataset is predicted where a different label appears in another dataset. Table 6.10 shows one example, using a model trained on SciER and evaluated on the GSAP-ERE development set.

Dataset	Original	unmapped	Kept
GSAP-ERE	34,902	131	10,809
SciER	11,501	0	11,492
SciNLP	1,648	1	1,366

Table 6.8: Relation retention breakdown across the unification pipeline. The difference between original and kept relations can also result from relations attached to dropped entities or from relation types outside the unified schema.

Dataset	Dropped entity labels
GSAP-ERE	DatasetGeneric, ReferenceLink, URL
SciER	none
SciNLP	Metric

Table 6.9: Entity labels removed because they have no stable counterpart in the unified schema.

Figures 6.4 and 6.5 illustrate how much schema detail is lost when projecting the original datasets into the unified schema. They show that GSAP-ERE represents more fine-grained model and data relations than SciER and SciNLP, which motivates preserving original schemas in the multi-head setting.

The confusion matrices reveal consistent alignment patterns. For example, GSAP-ERE labels such as MLModel, ModelArchitecture, and MLModelGeneric are frequently predicted as Method by a SciER-trained model. This supports mapping them to Method. Similarly, DataSource is frequently predicted as Dataset, supporting its collapse into Dataset. In contrast, ReferenceLink, URL, and DatasetGeneric show weak alignment and are often mapped to NIL. These labels are therefore excluded from the unified schema.

Based on these patterns, we select mappings by choosing the shared label with the strongest and most stable overlap across dataset pairs. Entity types without consistent alignment are excluded. This results in a minimal shared entity inventory consisting of Task, Dataset, and Method.

Relation Mapping Methodology. Relation alignment follows the same principle as entity alignment, but is conditioned on the unified entity mapping. This is necessary because relation labels depend on their subject and object types. We therefore compare relation signatures rather than relation labels alone.

After applying the entity mapping, we train supervised relation extraction models for each dataset using the original training splits. Each model is then evaluated on the development sets of the other datasets. Gold and predicted relations are represented as signatures of the form *(subject type, relation label, object type)*. This makes relation correspondences visible even when label names differ.

For every ordered pair of datasets, we construct confusion matrices over relation signatures. Relations are matched by selecting label pairs with stable overlap across dataset pairs. Directionality differences are resolved by inspecting subject and object types. Relations with reversed domain and range are inverted before matching. Semantically undirected relations, such as coreference and isComparedTo, are normalized

SciER model → GSAP gold ↓	Dataset	Method	Task	NIL
Task	13	16	**256**	28
Dataset	**388**	44	5	15
DatasetGeneric	190	54	188	**1139**
DataSource	**75**	4	3	21
MLModel	2	**119**	0	4
MLModelGeneric	6	284	15	**873**
Method	44	**721**	182	471
ModelArchitecture	12	**659**	16	82
ReferenceLink	27	13	2	**712**
URL	**16**	4	1	13
NIL	71	**129**	79	0

Table 6.10: Entity confusion matrix for a model trained on SciER and evaluated on the GSAP-ERE development set. Rows correspond to GSAP-ERE gold labels and columns to predicted labels. The highest value per row is highlighted in bold.

to a left-to-right order based on textual position.

To reduce sparsity, we also allow partial matching of relation signatures during analysis. When exact matches are rare, we consider matches where either the subject type or the object type aligns, provided the remaining components are compatible under the entity mapping. This helps identify stable relation correspondences despite differences in annotation density and schema design.

The resulting relation mapping defines a small set of shared relations supported by all three datasets. Relations without stable alignment are excluded from the unified schema. This includes relations tied to removed entity types and relations whose semantics remain dataset-specific.

6.8.3 Dataset Analysis Details

Corpus Metadata and Companion Application. We complement the dataset statistics in the main paper with metadata and schema views in the Dataset Explorer web application. The application provides paper-level views of corpus composition, publication years, outlets, data models, and prediction results. It is available at `https://huggingface.co/spaces/paper938487/companion-app`.

The metadata views show that the datasets are not sampled from the same publication distribution. SciNLP is centered on ACL/NLP papers and spans a long historical period. SciER covers broader AI topics and contains stronger computer-vision components. GSAP-ERE covers a wider ML and applied-ML space and has more heterogeneous source channels.

Figure 6.3 provides the temporal view behind the SciNLP density discussion in the main paper. It shows that SciNLP spans a long time period but has uneven yearly coverage and varying annotation density across entity types.

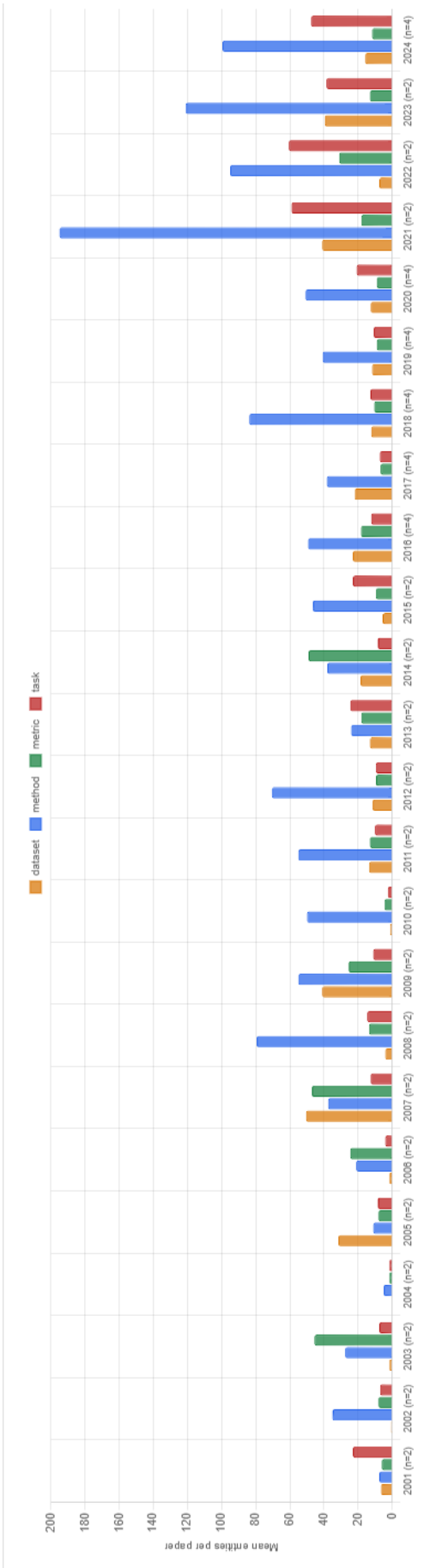


Figure 6.3: Temporal annotation-density view for SciNLP in the companion application. Bars show the mean number of entity mentions per paper by publication year and entity type. The view shows that SciNLP spans a long historical period, but its papers are unevenly distributed across years and annotation density varies substantially over time.

Entity Label Alignment Table 6.11 shows the entity label mapping used for unified evaluation. The mapping keeps only the shared conceptual core across all three datasets. This core consists of Task, Dataset, and Method.

The mapping collapses several GSAP-ERE labels into the shared inventory. For example, MLModel, MLModelGeneric, and ModelArchitecture are mapped to Method. DataSource is mapped to Dataset. Labels without stable counterparts, such as DatasetGeneric, ReferenceLink, and URL, are removed.

The mapping also removes Metric from SciNLP. Although metrics are important for NLP evaluation, this label is not supported across all three datasets. Keeping it would make unified evaluation depend on information available in only one source dataset.

Unified	**SciER**	**SciNLP**	**GSAP-ERE**	**Notes**
Task	TASK	Task	Task	Shared entity type across all three datasets.
Dataset	DATASET	Dataset	Dataset, DataSource	DataSource is mapped to Dataset.
Method	METHOD	Model	Method, MLModel, ModelArchitecture, MLModelGeneric	Model-related GSAP-ERE labels are mapped to Method.
DatasetGeneric	–	–	DatasetGeneric	GSAP-ERE-only label removed from the unified schema.
Metric	–	Metric	–	SciNLP-only label removed from the unified schema.
ReferenceLink	–	–	ReferenceLink	GSAP-ERE-only label removed from the unified schema.
URL	–	–	URL	GSAP-ERE-only label removed from the unified schema.

Table 6.11: Entity label alignment used for unified evaluation.

Relation Label Alignment Relation alignment is more difficult than entity alignment because relation labels interact with their subject and object types. Relation labels also differ in directionality and granularity across datasets. For this reason, relation mapping is derived over relation signatures of the form *(subject type, relation label, object type)* rather than over labels alone.

Table 6.12 shows the relation alignment used for unified evaluation. The unified relation set is organized into four semantic groups: TaskBinding, DataUsage, PeerRelations, and ModelDesign. Relations whose direction differs across datasets are inverted before evaluation. Symmetric relations such as comparison and coreference are normalized to a deterministic left-to-right order.

The unified mapping merges relations that encode similar extraction targets. For example, dataset training and evaluation relations are collapsed into a shared data-usage relation. Hierarchy and part-whole relations are collapsed into isHyponymOf. Relations that depend on removed entity types, such as metric-specific or reference-specific relations, are not part of the unified schema.

Semantic Group	**Unified**	**SciER**	**SciNLP**	**GSAP-ERE**	**Example sentence**
TaskBinding	appliedTo	Used-For	UsedFor	appliedTo	*We use BERT for named entity recognition.*
TaskBinding	benchmarkFor	Benchmark-For	evaluatedOn (inverted)	benchmarkFor	*We report results on SQuAD for question answering.*
DataUsage	trainedEvaluatedOn	Trained-With; Evaluated-With	trainedWith	trainedOn; evaluatedOn	*The model is trained and evaluated on MNLI.*
PeerRelations	coreference	Synonym-Of	similarWith	coreference	*We refer to BERT-base as BERT in the rest of the paper.*
PeerRelations	isHyponymOf	SubClass-Of; SubTask-Of	subclassOf; subtaskOf; partOf	isHyponymOf; isPartOf	*Sentiment analysis is a type of text classification.*
PeerRelations	isComparedTo	Compare-With	compareWith	isComparedTo	*We compare RoBERTa with BERT on the same benchmark.*
ModelDesign	usedFor	Part-Of; Used-For if object is Method	enhancedBy (inverted); Used-For if object is Method	architecture (inverted); isBasedOn (inverted)	*Our approach builds on a Transformer encoder.*

Table 6.12: Relation label alignment used for unified evaluation. Example sentences are illustrative and are not taken from the datasets.

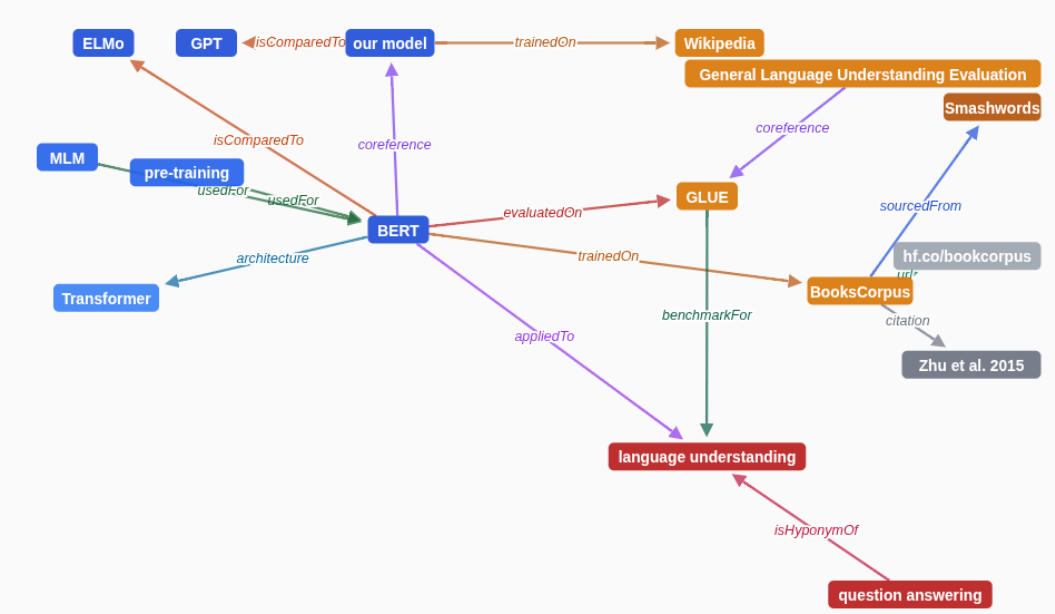


Figure 6.4: Toy example illustrating the expressiveness of the GSAP-ERE data model.

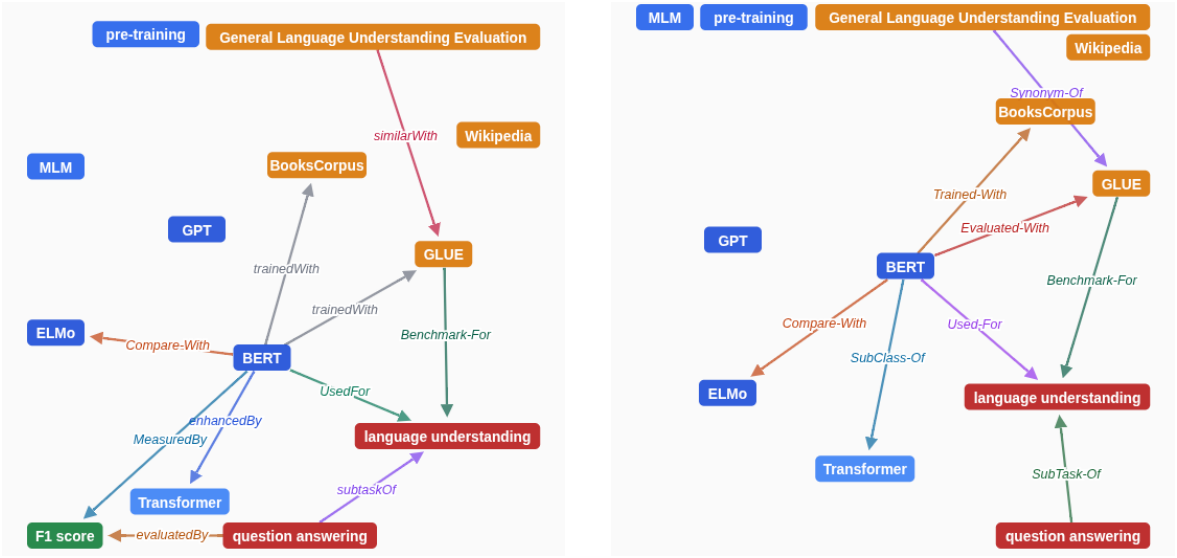
Schema Preview Examples Figures 6.4 and 6.5 illustrate the different schema expressiveness of the three datasets. The examples show why a unified schema is useful for controlled comparison but cannot preserve the full information content of all source datasets. GSAP-ERE contains the most detailed schema, while SciER and SciNLP use smaller but more compact inventories.

6.8.4 Additional Experimental Results

Corrected SciNLP Reference Scores The main paper uses the corrected arXiv version of SciNLP as the reference for reported HGERE scores. The corrected values reduce the earlier ACL scores from 94.15 to 79.53 for NER, from 61.74 to 49.28 for RE, and from 60.76 to 47.64 for RE+. These changes correspond to drops of 14.62, 12.46, and 13.12 F1.

Per-Label Cross-Dataset Entity Results Table 6.13 reports per-label entity cross-dataset evaluation results. The table complements the aggregate results in the main paper and shows that entity transfer differs by label. Dataset and Task generally transfer more consistently than Method, especially when evaluated across datasets with different model and method conventions.

Per-Label Cross-Dataset Relation Results. Table 6.14 reports per-label relation cross-dataset evaluation results. The results show stronger transfer degradation for relations than for entities. This supports the main-paper finding that unified relation labels remain affected by source-specific annotation semantics.



(a) SciNLP toy example. (b) SciER toy example.

Figure 6.5: Toy examples illustrating the schema expressiveness of SciNLP and SciER.

6.8.5 Auxiliary Transfer under a SciER NLP Domain Shift

We add a controlled auxiliary-transfer experiment to analyze how GSAP-ERE auxiliary data affects SciER performance under an explicit domain shift. Unlike the main experiments, this setup does not aim to reproduce the full SciER gains observed under the original split. Instead, we resplit SciER by domain and test whether different types of GSAP-ERE auxiliary data help when transferring from non-NLP SciER documents to NLP SciER documents.

We split documents into NLP and non-NLP groups using the document clusters shown in Table 6.15. Cluster 1 is treated as the NLP domain. Cluster 0 is treated as the non-NLP domain. For SciER, all non-NLP documents are assigned to the source side and all NLP documents are assigned to the target side. The resulting SciER split contains 66 non-NLP documents for training, 10 non-NLP documents for development, and 24 NLP documents for testing. No SciER NLP document is used during training or model selection.

For the GSAP-ERE auxiliary data, we use documents from the original GSAP-ERE train, development, and test splits. This is possible because GSAP-ERE is not evaluated in this experiment. It is used only as auxiliary supervision for the shared representation layers. There are 46 available non-NLP GSAP-ERE documents, so we use 46 documents in each auxiliary condition. The NLP auxiliary condition uses 46 NLP GSAP-ERE documents. The non-NLP auxiliary control uses 46 non-NLP GSAP-ERE documents. The mixed auxiliary control uses 23 NLP and 23 non-NLP GSAP-ERE documents.

All runs use the same multi-head HGERE architecture as in the main experiments (Yan et al., 2023). The shared encoder and relation representations are trained on the active training input. The SciER and GSAP-ERE prediction heads remain separate. The SciER head is used for model selection on the SciER non-NLP development set and final evaluation on the SciER NLP test set. The GSAP-ERE head is used only to provide auxiliary training signal.

Label	Train → Test ↓	GSAP	SciER	SciNLP
TASK	GSAP	*76.6*	48.3	**51.0**
	SciER	**84.2**	*89.8*	76.5
	SciNLP	**60.3**	51.9	*82.6*
DATASET	GSAP	*83.8*	**67.3**	63.5
	SciER	**91.4**	*92.9*	83.8
	SciNLP	**67.9**	60.9	*85.1*
METHOD	GSAP	*87.6*	**65.4**	62.9
	SciER	**88.0**	*95.0*	**88.0**
	SciNLP	**61.2**	57.7	*89.7*
micro	GSAP	*86.3*	**63.5**	61.8
	SciER	**87.8**	*93.8*	85.3
	SciNLP	**61.9**	56.8	*87.3*

Table 6.13: Entity cross-dataset evaluation results on the test splits of the three datasets. Scores are F1 in percent. Italic values mark same-dataset evaluation. Bold values mark the best cross-dataset score in each row.

This experiment distinguishes generic auxiliary transfer from domain-matched auxiliary transfer. Generic auxiliary transfer means that adding GSAP-ERE improves SciER performance independently of whether the added documents are NLP-related. Domain-matched auxiliary transfer means that NLP-related GSAP-ERE documents improve the SciER NLP target more than a size-matched non-NLP auxiliary subset.

We evaluate these effects through pairwise comparisons between the four runs. The generic auxiliary effect is measured by comparing each auxiliary condition against the SciER-only baseline. The domain-matched auxiliary effect is measured by comparing the NLP auxiliary condition against the non-NLP auxiliary control. The mixed auxiliary condition tests whether a balanced combination of NLP and non-NLP auxiliary documents is sufficient to obtain the same benefit as the NLP-only condition.

Because this experiment evaluates only 24 SciER NLP documents, we interpret small differences cautiously. We report mean and standard deviation over multiple random seeds. We also report the number of papers, sentences, entity mentions, relation mentions, and NIL relation candidates for each subset. The hypotheses are considered supported only if the relevant improvements are stable across seeds and visible on the primary relation extraction metrics.

Table 6.18 shows that all auxiliary conditions improve SciER target-domain NER compared with the SciER-only baseline. The baseline reaches 84.2 ± 2.6 NER F1 on the NLP test set, while the auxiliary runs reach between 87.9 and 88.4 F1. The best NER result is obtained by the mixed auxiliary condition, which improves over the baseline by 4.2 F1 points. The non-NLP and NLP auxiliary conditions are nearly tied on the target test set, suggesting that the NER gain is not specific to NLP auxiliary documents. The auxiliary runs also reduce the dev–test gap from -6.2 to between -4.9 and -4.1, indicating more stable transfer from the non-NLP development domain to the NLP target domain.

Table 6.19 shows a weaker and less stable effect for relation extraction. The baseline

Label	Train → Test ↓	GSAP	SciER	SciNLP
benchmarkFor	GSAP	*28.6*	**29.4**	19.2
	SciER	**41.4**	*65.5*	34.4
	SciNLP	46.2	**56.0**	*66.7*
appliedTo	GSAP	*51.7*	**34.9**	21.2
	SciER	**51.9**	*69.2*	40.4
	SciNLP	**40.0**	32.0	*54.9*
trainedEvalOn	GSAP	*59.0*	**41.3**	32.1
	SciER	**40.2**	*69.2*	32.6
	SciNLP	48.3	**53.8**	*56.0*
isHyponymOf	GSAP	*36.2*	**26.2**	8.8
	SciER	**28.5**	*57.0*	17.4
	SciNLP	**17.8**	14.6	*35.3*
isComparedTo	GSAP	*45.9*	**26.9**	15.3
	SciER	**53.3**	*72.9*	49.1
	SciNLP	54.0	**72.1**	*57.1*
coreference	GSAP	*71.2*	**43.7**	30.0
	SciER	**59.2**	*91.4*	43.1
	SciNLP	27.9	**36.8**	*54.5*
usedFor	GSAP	*59.3*	**15.8**	0.8
	SciER	**40.4**	*59.2*	0.7
	SciNLP	**3.9**	0.0	*31.6*
micro	GSAP	*56.2*	**26.7**	12.5
	SciER	**45.3**	*67.6*	30.4
	SciNLP	32.8	**39.3**	*51.6*

Table 6.14: Relation cross-dataset evaluation results on the test splits of the three datasets. Scores are F1 in percent. Italic values mark same-dataset evaluation. Bold values mark the best cross-dataset score in each row.

reaches 54.2 ± 2.1 RE F1 on the NLP test set. The non-NLP auxiliary condition improves this to 56.7 ± 2.6, and the mixed auxiliary condition obtains the best score with 57.5 ± 1.3. In contrast, the NLP auxiliary condition reaches only 54.3 ± 7.6 on the target test set, despite having the highest development score. This result suggests that NLP auxiliary data alone does not yield reliable RE transfer in this setup.

The development scores show that source-domain model selection does not reliably predict target-domain performance. This is most visible for RE, where the NLP auxiliary condition has the highest development score at 69.1 ± 1.3 but does not improve over the baseline on the NLP test set. The mixed auxiliary condition has a lower development score but the best and more stable target-domain RE score. This gap indicates that the non-NLP development set favors models that fit the source domain, while target-domain relation extraction depends on auxiliary signals that transfer beyond the selected development domain.

Overall, the experiment supports a limited auxiliary-transfer effect under the forced

	GSAP-ERE			SciER			SciNLP		
Cluster	train	dev	test	train	dev	test	train	dev	test
NLP	44	4	5	19	2	3	49	6	3
non-NLP	35	6	5	61	8	7	1	0	0

Table 6.15: Document counts by cluster and original split. Cluster 1 is treated as the NLP domain and Cluster 0 as the non-NLP domain.

SciER domain-shift split	Domain	Docs	Original split composition	Purpose
Train	non-NLP	66	61 train + 5 test	source-domain training
Dev	non-NLP	10	8 dev + 2 test	source-domain model selection
Test	NLP	24	19 train + 2 dev + 3 test	target-domain evaluation

Table 6.16: Resampled paper-wise SciER split for the NLP domain-shift experiment. All NLP documents are assigned to the target-domain test set, while non-NLP documents are used for training and development. The original SciER split labels are shown only to report document provenance.

non-NLP → NLP shift. For NER, adding GSAP-ERE supervision improves target-domain performance regardless of the auxiliary domain. For RE, the best result comes from combining NLP and non-NLP auxiliary documents, while the NLP-only condition is unstable across seeds. We therefore interpret the gain as a general representation-sharing effect that is strongest when the auxiliary data remains diverse, rather than as a pure domain-matching effect. Because the target test set contains only 24 SciER NLP documents, small differences should be treated cautiously, especially for RE.

Run	Description	Train			Dev	Test
		SciER	GSAP-ERE auxiliary		SciER	SciER
		non-NLP	non-NLP	NLP	non-NLP	NLP
A	SciER non-NLP baseline	66	0	0	10	24
B	NLP auxiliary	66	0	46	10	24
C	non-NLP auxiliary control	66	46	0	10	24
D	mixed auxiliary control	66	23	23	10	24

Table 6.17: Experimental conditions for the SciER NLP domain-shift experiment. All runs use the same SciER non-NLP → NLP split. GSAP-ERE is used only as auxiliary training data and has no development or test role in this experiment. The auxiliary conditions use 46 GSAP-ERE documents, which is the maximum size that allows a balanced comparison between NLP and non-NLP auxiliary documents.

Run	Description	Dev F1	Test F1	Δ test–dev
A	SciER baseline	90.4 ± 0.7	84.2 ± 2.6	−6.2
B	non-NLP auxiliary	92.8 ± 0.1	87.9 ± 0.2	−4.9
C	NLP auxiliary	92.8 ± 0.5	88.0 ± 2.5	−4.9
D	mixed auxiliary	92.5 ± 0.4	88.4 ± 0.5	−4.1

Table 6.18: NER results for the SciER non-NLP → NLP domain-shift experiment. Scores are micro-F1 means and standard deviations over five seeds.

Run	Description	Dev F1	Test F1	Δ test–dev
A	SciER baseline	64.7 ± 0.7	54.2 ± 2.1	−10.5
B	non-NLP auxiliary	68.2 ± 0.8	56.7 ± 2.6	−11.5
C	NLP auxiliary	69.1 ± 1.3	54.3 ± 7.6	−14.8
D	mixed auxiliary	68.0 ± 0.5	57.5 ± 1.3	−10.5

Table 6.19: RE results for the SciER non-NLP → NLP domain-shift experiment. Scores are micro-F1 means and standard deviations over five seeds.

7

Exploration and Analysis of Extracted Machine Learning Knowledge

Publication Information

Title: Methods Miner: Exploring Machine Learning Methods, Tasks, and Datasets in Scholarly Publications

Authors: Wolfgang Otto, Sharmila Upadhyaya, Lu Gan, Kanishka Silva, Stefan Dietze

Status: To be submitted as demo paper

Reuse: This chapter reproduces the text of the publication listed above in thesis layout.

Context

This publication turns the thesis contributions into a user-facing workflow. While earlier chapters focus on data, schemas, and extraction methods, researchers often need tools that let them explore a focused set of papers and quickly find which methods, model architectures, tasks, and datasets appear in that collection, together with the sentence-level evidence. The Methods Miner system addresses this need by processing user-provided PDFs and exposing extracted entities and their contexts through an interactive interface.

The system integrates a PDF-to-text and text-structuring pipeline with the fine-grained scholarly NER model introduced earlier in the thesis. Extracted mentions are stored together with links to their original text units, so users can move from aggregated views (counts and document frequencies) to concrete usage contexts in one step. The chapter describes the end-to-end architecture, the main components of the processing pipeline, and a demonstration scenario that shows how the system supports exploration in practice.

In the thesis structure, this chapter is not only an application endpoint, but also a practical test of the evidence-linking principle developed in the earlier chapters. It shows why the earlier design choices matter: fine-grained entity types and sentence-linked contexts make extracted information inspectable and usable for analysis. It also provides a platform that can be extended with relation extraction models from GSAP-ERE and with cross-dataset insights from the schema-alignment chapter (6), moving the thesis from extraction results to practical use.

7.1 Introduction

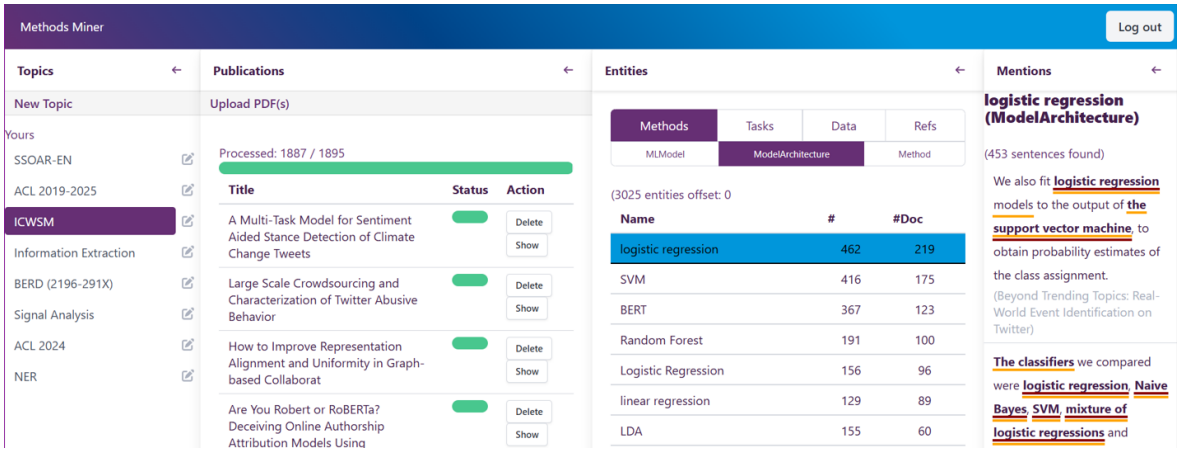


Figure 7.1: Screenshot of the Methods Miner.

Recent work has shown that it is often difficult to reproduce ML-related research and to understand which models and datasets are state-of-the-art for specific tasks (Raff, 2019; Pineau et al., 2021; Ferrari Dacrema et al., 2019).

These problems are amplified by the rapid growth of ML publications, which hinders individual researchers keeping track of methods, datasets, and tasks and their relations to publications in their own field. Recent developments in LLMs have been resorted to assist academic paper search (He et al., 2025), literature review (Kasanishi et al., 2023) and research paper summarization (Dai et al., 2024). However, the state-of-the-art vanilla LLMs still lag behind fine-tuned PLMs significantly in scholarly entity and relation extraction (Otto et al., 2026).

This gap is especially clear for named entity recognition in our setting. On GSAP-ERE (Otto et al., 2026), few-shot / zero-shot prompting with LLMs performs substantially worse than a supervised fine-tuned model for scholarly NER (e.g., 44.4 vs 80.6 micro-f1 score), indicating that prompting alone is not sufficient for reliable extraction. Similar results on SciER (Q. Zhang et al., 2024) and SciNLP (Duan et al., 2025) suggest that prompting is not yet a dependable substitute for fine-tuned scholarly NER, motivating our focus on exposing a fine-tuned model in the demo.

While LLMs are accessible almost everywhere, specialized fine-tuned PLMs for scholarly extraction are not. Exposing such a model through a public web interface, including API access, is therefore a central goal of our system. In practice, researchers still need to scan many papers manually to answer simple questions such as which model architectures are common for a given task or which datasets are frequently used in a topic area. Scholarly Information Extraction (IE) can support this process by turning unstructured scientific text into structured representations (Luan et al., 2018; Jain et al., 2020; Q. Zhang et al., 2024).

Named entity recognition (NER) and relation extraction (RE) are central steps in this direction and form the basis for downstream tasks such as scholarly knowledge graph construction and analytic tools for research monitoring (Dessì et al., 2022; Dasigi et al., 2021). However, many existing systems and datasets are designed for large-scale, centralized pipelines (Dessì et al., 2022) and do not directly address the everyday

workflow of researchers who want to explore a focused set of publications, such as their own work or a curated topic collection.

In this demo paper, we present *Methods Miner*, a web-based system that helps researchers explore machine learning methods, tasks, and data in their own research domains. The tool accepts user-provided PDFs, runs a background processing pipeline based on Grobid and a scholarly NER component (GSAP-NER (Otto et al., 2023)), and stores text units and entity mentions in a structured database. The frontend, shown in Figure 7.1, provides four panels for topics, publications, entities, and sentence-level contexts, allowing users to move from high-level overviews to concrete evidence in the text. The underlying architecture is summarized in Figure 7.2 and combines a Vue.js interface with a GraphQL backend, a Celery-based processing layer, and a Postgres data store. Our contributions are:

- We present a web-based system that turns user-selected ML publications into an overview of methods, model architectures, tasks, and datasets used in a given research domain.
- We integrate a PDF processing and text structuring pipeline with a fine-grained scholarly NER model (GSAP-NER), enabling entity-level exploration at sentence granularity.
- We provide an interactive interface that organizes results by topics, documents, entity types, and contexts, supporting researchers in inspecting both overall trends and concrete usages in their own publication sets.

7.2 System Description

The analysis of scientific publications takes time and requires solid computing capacity. To keep the system usable while these steps run, the processing is handled in the background. As shown in Figure 7.2, the frontend sends uploads to the backend through the GraphQL API, which passes new tasks to the Celery workers. These workers handle the full extraction pipeline, including document ingestion, entity recognition, linking, and context extraction. Since multiple workers can run at the same time, the system can process many publications in parallel. The extracted information is stored in the database in a structured way, which supports fast access to publications, entities, and mentions in the user interface. This setup allows users to explore results without being slowed down by ongoing processing tasks and ensures that the system can scale when the number of uploaded documents increases.

7.2.1 Frontend Implementation

The frontend provides access to research topics, publications, extracted entities, and their mentions, as shown in Figure 7.1. It is implemented as a web application using vue.js[1]. Data is retrieved through Apollo[2], which sends GraphQL queries to the backend and keeps the interface updated when new extraction results become available.

[1] We used version 3.5.13 `https://vuejs.org/`

[2] `https://www.apollographql.com/`

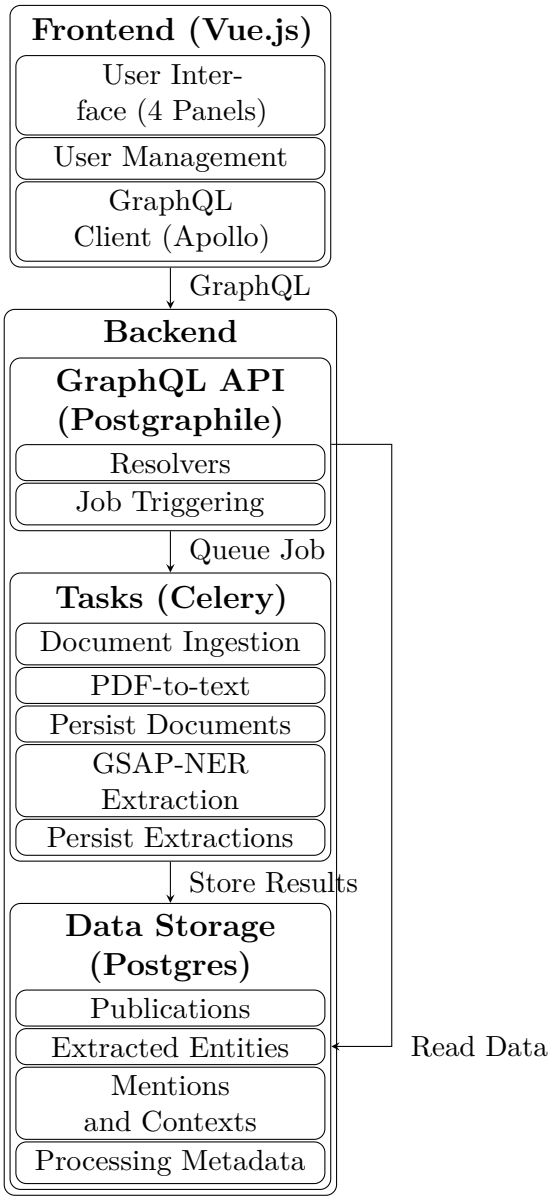


Figure 7.2: Top-down system overview of Methods Miner.

The user interface is divided into four panels shown in Figure 7.1. The leftmost panel lists user-defined topics. Selecting a topic loads the related publications, which appear in the second panel. After uploads are processed, the third panel displays extracted mentions grouped into the main entity types Method, Task, Data, and References, along with their subtypes. The fourth panel shows the sentence-level context for any selected mention, which supports direct inspection of the extracted information. The frontend communicates with the backend through GraphQL and depends on the backend to provide processed publications, extracted entities, and progress updates. Figure 7.2 shows how both parts work together: the frontend sends queries and uploads, while the backend handles processing tasks and returns the stored results. The following section describes the backend components supporting this data flow.

7.2.2 Service and Data Backend

The backend handles data access, task management, and communication with the database, as shown in Figure 7.2. All data is stored in a Postgres database and exposed through a GraphQL API generated with Postgraphile. The API receives uploads from the frontend, creates new processing jobs, and returns stored extraction results and is directly accessible to users as well. Background processing is handled by a Celery-based task system. When new documents are uploaded, the API writes a job entry to the database and adds the task to the Celery queue. Multiple workers can run in parallel, which enables process many publications at the same time. Progress and error states are also written back to the database so the frontend can show updates while the extraction is still running. The pipeline can be extended by adding new tasks, for example for relation extraction or other analysis steps.

Processing Pipeline

The processing pipeline prepares uploaded publications for later extraction steps. When a new document is received, the system first runs Grobid to extract structured text from the PDF. This includes the main body text, section headings, paragraph boundaries, sentence splits, and the reference section. All extracted text units are stored in the database as spans with their positions in the original document. Each span points back to a single plain-text field that contains the full publication in a readable form without annotations. This layout allows to work with different levels of document, such as sentences, paragraphs, or sections. After the text units are stored, each paragraph is sent as a separate chunk to the NER component. The identified mentions returned by the NER system are then inserted into the database and linked to the corresponding text spans.

NER Component

The NER component identifies mentions of machine learning methods, models, tasks, and data in the processed documents. It is based on the entity schema and the model introduced in (Otto et al., 2023), which can detect ten scholarly entity types. Each paragraph from the processing pipeline is sent to the NER model as a separate input. The model returns the detected mentions together with their spans. These mentions are then stored in the database and linked to the corresponding text units, making it possible to access the exact sentence or paragraph where the mention occurs.

This setup allows the system to provide structured information about the machine learning content of each publication. For example, the table in Figure 7.1 shows the most frequent mentions of the type *ModelArchitecture* in a corpus of papers from ICWSM. These results help users examine how methods, models, tasks, and datasets appear across the uploaded papers.

7.3 Demonstration Scenario

We demonstrate how the system creates an overview of machine learning methods, tasks, and data used in a user's specific research domain. The workflow starts with a set of publications provided by the user as PDF files. After the upload, the processing

pipeline runs in the background and extracts the structured text, text units, and the relevant mentions. Once the extraction is complete, the frontend presents the results in the four panels shown in Figure 7.1.

Upload and Processing

The user begins by selecting a topic and uploading several publications related to that topic. Each uploaded PDF is processed as described in Section 2, where the text extraction, span creation, and NER steps run asynchronously. When the extraction finishes, the system updates the interface silently.

Exploring the Extracted Entities

The Entity Panel organizes all detected mentions by their types. This includes the method-related types *MLModel*, *ModelArchitecture*, and *Method (misc)*, the task type *Task*, and the data-related types *Dataset* and *DataSource*. For each type, the interface shows both the total number of mentions and the document frequency. This gives a quick overview of which methods, models, tasks, and datasets appear in the uploaded publications.

Inspecting Mentions in Context

Selecting an entity in the Entity Panel loads all sentence-level contexts in the final panel. This allows the user to check how the entity is used in the publication and to trace it back to the exact location in the text. As each mention is linked to its original text span, the user can move from the list of entities to the corresponding sentences with one step.

Deployment Note

At the time of writing, the public demo is hosted on a single GPU server running Ubuntu 24.04 LTS with 128 CPU cores, 1.37 TB RAM, one NVIDIA H100 (90 GB), and a 500 GB SSD. The deployment is intended for interactive use and evaluation; latency depends mainly on input document length and request concurrency, and scaling would require a setup with additional GPU workers for a replicated inference service behind the API.

7.4 Related Work

Our work connects to research on machine learning (ML)–focused scholarly IE, and systems for interactive exploration of extracted data. A large body of work studies information extraction (IE) from scientific articles and surveys main tasks, datasets, and methods (Nasar et al., 2018; Y. Yang et al., 2022; Z. Zhang et al., 2025). These surveys cover classical pipeline architectures as well as recent deep learning and generative approaches for IE. More recently, large language models have been used for structured extraction at scale from scientific text, for example to build structured databases in materials science (Dagdelen et al., 2024). MASSW (X. Zhang et al., 2025) uses LLMs to summarize five research workflow aspects across 152k scientific papers. Compared to

these general surveys and large-scale pipelines, our system focuses on a narrower but practically important use case: fine-grained ML method, task, and data information in user-defined publication collections. Several tools extract structured information from PDF articles, especially on the document layout and bibliographic level. GROBID (Lopez, 2009) and CERMINE (Tkaczyk et al., 2015) extract headers, references, and document structure from scientific PDFs and are widely used as preprocessing components in digital libraries and text-mining systems. Large corpora such as S2ORC (Lo et al., 2020) provide machine-readable full text and metadata for tens of millions of articles and are a key resource for downstream IE tasks. Shared tasks such as ScienceIE (Augenstein et al., 2017) target keyphrase and relation extraction from scientific publications. Methods Miner builds on this line of work by using GROBID extraction to obtain structured text units, but it targets a more specialized schema for ML research and is designed for interactive analysis on user-provided corpora instead of global web-scale collections. There is also growing work on metadata models and knowledge representation for ML experiments and artifacts. Gesese et al. survey metadata for ML models and datasets and discuss how existing schemas can support reproducibility and discovery (Gesese et al., 2025). Our approach is complementary: rather than defining a new metadata model, Methods Miner extracts text-based mentions and their local contexts, which can serve as input signals for populating such metadata schemas or for building knowledge graphs over ML research. Finally, several recent systems combine IE with interactive user interfaces for structured data exploration from scientific text. SciDaSynth (X. Wang et al., 2025), for example, uses large language models to build task-specific tables from scientific literature and provides interactive workflows for validation and refinement. Methods Miner follows a similar idea of supporting interactive analysis but focuses on ML publications, uses supervised scholarly IE models instead of general-purpose prompting, and organizes the user interface around entity types (methods, tasks, data) and their document-level distributions, as shown in Figures 7.1 and 7.2.

7.5 Conclusion and Future Work

This paper presented Methods Miner, a system for exploring machine learning methods, tasks, and data in user-defined research domains. The tool processes uploaded PDF publications, extracts structured text units, and applies a scholarly NER component to identify ML-related entities. The extracted information is stored in a database and exposed through a web-based interface built on Vue.js and GraphQL, allowing users to inspect entity distributions and sentence-level contexts across their own publication sets. We described the overall architecture, the processing pipeline based on Grobid and GSAP-NER, and a demonstration scenario that shows how researchers can use the system to obtain an overview of ML techniques used in their field.

In future work, we plan to extend the extraction pipeline with the GSAP-ERE relation model to capture relations between methods, models, tasks, and data. This will require a document-level component that combines relation extraction with coreference resolution and corpus-wide entity linking. We aim to link mentions within a single paper, across papers in the corpus, and to external knowledge bases. These extensions will give the system a clearer knowledge graph perspective and will enable more advanced exploration and querying of ML-related information in scientific publications.

Ethical Use of Data and Informed Consent

This work is based solely on published scientific articles provided as PDF files by users and on system-level processing metadata. No experiments with human participants were conducted, and no individual-level personal or sensitive data were collected, stored, or analyzed. Therefore, formal ethics board review and informed consent were not required under the applicable institutional and national guidelines. The work complies with ACM's Publications Policy on Research Involving Human Participants and Subjects.

8

Conclusion and Future Work

This chapter summarizes the results of the thesis, answers the research questions, discusses the main limitations, and outlines future work. The chapter follows the cumulative structure of the thesis: fine-grained extraction from full text, robustness and comparability across extraction settings, and user-facing analysis of extracted machine-learning knowledge.

8.1 Summary of Contributions

This thesis studied how machine-learning knowledge in scholarly publications can be extracted as fine-grained, evidence-linked, and comparable records. It started from the observation that the information needed to understand ML experiments is distributed across full papers, expressed through heterogeneous artifact mentions, and shaped by the representational view imposed by an annotation schema, extraction model, or downstream application. The thesis therefore did not treat scholarly information extraction only as span detection. Instead, it connected entity extraction, relation extraction, schema-aware comparison, and user-facing exploration into one cumulative line of work.

The first contribution is the entity-level representation developed in Chapter 3. GSAP-NER defines a fine-grained schema for ML-related entities and concepts, including concrete models, model architectures, datasets, data sources, tasks, methods, and generic artifact mentions. By annotating full-text publications rather than only abstracts or selected sentences, the chapter shows that ML artifact mentions occur in many parts of a paper and require distinctions that are lost in coarser scholarly IE schemas. The baseline experiments show that supervised language models can learn this representation, but they also show that conceptual and context-dependent entity types remain more difficult than concrete named artifacts. This establishes the entity layer on which the later relation and exploration work builds.

The second contribution is the LLM-assisted extraction perspective developed in Chapter 4. This chapter studies software-related scholarly IE in a shared-task setting and reformulates parts of the extraction problem as single-choice question answering. It shows that retrieval-supported prompting and task reformulation can improve extraction in limited-supervision settings, especially when the prompt is constrained by known entity and relation signatures. At the same time, the chapter reinforces a broader thesis point: flexible LLM-based extraction still depends on controlled schemas, reliable evidence, and evaluation procedures that make outputs comparable to gold annotations. It therefore complements the supervised extraction line rather than replacing it.

The third contribution is the relation-level representation developed in Chapter 5. GSAP-ERE extends the entity layer from isolated artifact mentions to typed relations between ML artifacts. The chapter models links such as model design, task binding, data usage, data provenance, peer comparison, and external referencing. This move is central to the thesis because knowing that a paper mentions a model, a dataset, and a task is not sufficient for reconstructing an experimental workflow. The relation layer turns entity mentions into sentence-level evidence for how artifacts are trained, evaluated, compared, applied, sourced, or documented.

The fourth contribution is the cross-dataset generalization study in Chapter 6. This chapter moves beyond single-resource evaluation and asks how scientific ERE models behave when datasets overlap in broad target domain but differ in schemas, annotation density, corpus composition, and relation semantics. The unified evaluation setting makes controlled comparison possible, but the results show that schema unification alone is not enough for robust relation extraction transfer. The schema-preserving multi-dataset setup addresses this by sharing representations while keeping each dataset's original prediction space. This contribution clarifies a central thesis claim: scientific IE datasets are not only different label inventories, but different representational views over scholarly evidence.

The fifth contribution is the user-facing exploration system developed in Chapter 7. Methods Miner applies the extraction line to user-defined paper collections and exposes extracted ML-related entities through an interface for inspection and comparison. The system connects extracted records back to sentence-level contexts, so that users can move from aggregate views to the textual evidence behind them. This chapter demonstrates why evidence links are not only useful for evaluation, but also for downstream use. It closes the thesis line by showing how fine-grained extraction can support exploration of methods, tasks, datasets, and other ML-related artifacts across scholarly corpora.

Taken together, these contributions form a progression from representation to extraction, from extraction to cross-dataset comparison, and from comparison to use. The thesis contributes datasets, schemas, models, evaluation setups, and a prototype system, but its main result is the connection between these parts. Fine-grained ML-related information can be extracted from full-text scholarly publications, but its usefulness depends on preserving evidence, respecting schema differences, and exposing the resulting records in forms that support inspection and comparison. The next section summarizes these findings with respect to the research questions.

8.2 Answers to the Research Questions

This section returns to the research questions stated in Section 1.2. The thesis addressed how machine-learning artifacts and their relations can be extracted from full-text scholarly publications, how such extraction can be compared across scientific ERE datasets, and how extracted records can support inspection and analysis. The answers are summarized below at the level of the three main research questions and their sub-questions.

RQ-A: Extraction from full text. *RQ-A* concerns how machine-learning artifacts and their workflow relations can be extracted from full-text scholarly publications as fine-grained, evidence-linked records. The thesis answers this question by combining

the entity schema and corpus of GSAP-NER (Chapter 3) with the relation schema and corpus of GSAP-ERE (Chapter 5). Together, these resources show that ML-related scholarly IE can be modeled as a closed-schema full-text extraction task in which concrete models, model architectures, datasets, data sources, methods, tasks, generic mentions, and external references are treated as distinct evidence-bearing objects. The answer is therefore positive, but bounded: fine-grained extraction is feasible with supervised models and manually curated resources, while conceptual entity types, ambiguous model references, and relation extraction remain harder than detecting concrete named artifacts.

RQ-A1 asked which artifact distinctions are needed for ML papers and how they can be represented in a learnable and evaluable schema. The thesis answers this through the GSAP-NER entity inventory. The distinction between concrete ML models and model architectures is necessary because the same surface form can refer either to an executable artifact or to a conceptual design, depending on context. The distinction between datasets and data sources is also necessary because stable benchmark datasets and unstable sources, such as web platforms, play different roles in experimental workflows. The inclusion of generic mentions is important because ML papers often refer to artifacts through phrases such as "our model" or "the benchmark", which can be needed for later relation extraction, coreference, and evidence inspection.

RQ-A2 asked how fine-grained artifact and relation annotations can be created across full-text scholarly publications while preserving sentence-level evidence. The thesis addresses this question by annotating full-text ML publications in GSAP-NER and GSAP-ERE and by keeping entity and relation instances linked to their source spans and sentences. This design shows that full-text annotation is needed because evidence about models, datasets, tasks, comparisons, and provenance is not confined to abstracts. At the same time, the work also shows a limit of the chosen design: the corpora cover full papers, but the extraction targets remain sentence-level. Cross-sentence relations, document-level coreference, table evidence, figure evidence, and citation-graph signals remain outside the main extraction target of the thesis.

RQ-A3 asked how workflow relations between ML artifacts can be modeled and extracted as sentence-level relation evidence. The thesis answers this through GSAP-ERE, which extends the GSAP entity layer with typed relations for model design, task binding, data usage, data provenance, data properties, peer relations, and referencing. This relation layer shows that entity extraction alone is not sufficient for reconstructing ML experiments. A paper-level record needs to represent whether a model is trained on a dataset, evaluated on a benchmark, applied to a task, compared with another model, or linked to an external source. The experiments show that these relations can be learned by supervised ERE models, but also that fine-grained relation extraction remains substantially more difficult than entity extraction.

RQ-B: Robustness, comparability, and generalization. *RQ-B* asked how scientific ERE models can be compared and transferred across datasets that overlap conceptually but differ in schema design, annotation density, corpus selection, and relation semantics. The thesis answers this question in two steps. First, it introduces a unified schema and mapping procedure for SciER, SciNLP, and GSAP-ERE, which enables controlled cross-dataset evaluation. Second, it shows that unified labels are useful for comparison but insufficient as a training schema for robust relation extraction

transfer. The stronger answer is schema-preserving transfer: models should share representations where extraction signals overlap, but preserve dataset-specific prediction heads where annotation views differ.

RQ-B1 asked how scientific ERE datasets with different schemas can be aligned into a shared representation for controlled cross-dataset evaluation. The thesis answers this by defining a canonical alignment layer for entity and relation labels and by reporting how much of each dataset is retained or lost under this mapping. This makes cross-dataset evaluation possible because predictions and gold annotations can be compared in a shared output space. However, the answer also shows that alignment is not neutral. It can merge labels, drop labels, simplify directionality, and remove distinctions that individual datasets were designed to preserve.

RQ-B2 asked to what extent models trained on one scientific ERE dataset transfer to another when outputs are mapped into a shared schema. The thesis answers that transfer is limited and differs between entity extraction and relation extraction. Under unified labels, broad entity concepts transfer better than relation-level behavior, especially when partial span matching is considered. Relation extraction remains much less transferable because similar relation names can encode different evidence rules, argument choices, and annotation scopes. Thus, schema unification supports controlled measurement, but it does not by itself solve cross-dataset generalization.

RQ-B3 asked whether shared representations can improve scientific ERE across benchmarks when each dataset keeps its own original prediction space. The thesis answers this positively for some datasets and with an important asymmetry. The dataset-conditioned multi-head model improves SciER and SciNLP, especially for relation extraction, while GSAP-ERE benefits less and remains strongest when trained on its own fine-grained data. This shows that schema-preserving transfer can exploit shared extraction signals without forcing all datasets into a reduced label space. It also shows that larger or more fine-grained resources can provide useful auxiliary signal for less detailed schemas, while the reverse direction may provide less benefit.

RQ-C: Use and analysis at scale. *RQ-C* asked how extracted machine-learning knowledge can be organized and presented so that users can inspect evidence and analyze paper collections. The thesis answers this through Methods Miner, a prototype system that processes user-provided PDF collections, extracts structured text units, applies the GSAP-NER model, stores extracted mentions and contexts, and exposes them through a web interface. The answer is therefore based on demonstrated feasibility rather than on a full user study. The system shows that entity-level extraction can support topic-based paper organization, entity-type summaries, mention frequencies, document frequencies, and direct inspection of sentence-level contexts. It does not yet provide a complete solution for relation-based exploration, user correction, cross-paper entity linking, or robust normalization.

RQ-C1 asked which record structure and evidence-linking strategy supports verification of extracted records. The thesis argues that a practical record structure should store publications, structured text units, extracted mentions, entity types, offsets, and sentence-level contexts, thereby supporting navigation from aggregate views back to textual evidence. Methods Miner implements this strategy by connecting extracted mentions to sentence-level contexts in the interface. This supports inspection and verification, but not yet full correction workflows. Therefore, the thesis answers the

evidence-inspection part of RQ-C1, while interactive correction remains future work.

RQ-C2 asked how artifacts should be normalized and grouped across papers to support aggregation and comparison. The thesis answers this only at the mention-grouping level. Methods Miner groups extracted mentions by topic, publication, entity type, surface form, and context, which supports overview-style analysis of methods, tasks, and data mentions across a user-defined collection. However, reliable artifact-level aggregation across papers requires coreference resolution, entity linking, and normalization to canonical artifacts. These steps are identified as future work rather than fully solved in the current system.

RQ-C3 asked which analysis and exploration tasks are directly supported by artifact-and-relation extraction and where the limits of text-only coverage lie. The thesis answers that the current system directly supports entity-level exploration tasks: identifying which ML-related artifact mentions occur in a collection, comparing their frequency across papers, and inspecting the contexts in which they are used. The limits are also clear. Text-only and mention-level extraction cannot fully reconstruct experimental workflows when evidence is spread across tables, figures, appendices, references, or several sentences. A complete workflow-level system would need relation extraction, document-level coreference, entity linking, and support for non-textual evidence.

Overall, the research questions are answered in a cumulative way. *RQ-A* is answered by the fine-grained entity and relation resources and the supervised extraction baselines. *RQ-B* is answered by the unified and schema-preserving cross-dataset evaluation framework. *RQ-C* is answered as a prototype-based demonstration of evidence-linked exploration, with normalization, correction, and relation-centered exploration left as future work. The combined answer is that ML knowledge can be extracted from full-text scholarly publications as evidence-linked records, but its reliable reuse depends on preserving schema distinctions, keeping evidence inspectable, and making the limits of mention-level extraction explicit.

8.3 Limitations

The contributions of this thesis show that ML-related knowledge can be extracted from scholarly publications as fine-grained, evidence-linked records. However, the results should be interpreted within four main limits. These limits concern the corpus and domain scope, the dependence on annotation schemas, the granularity of textual evidence, and the relation between benchmark evaluation and practical usefulness.

Corpus and domain scope. The thesis focuses mainly on machine-learning-related scholarly publications and on selected scientific ERE datasets. GSAP-NER and GSAP-ERE are built around ML publications, and the cross-dataset experiments compare SciER, SciNLP, and GSAP-ERE. This focus is necessary for fine-grained artifact extraction, but it also limits how far the results can be transferred to other scientific domains, publication types, and writing styles. For example, survey papers, reproducibility studies, interdisciplinary papers, and papers with strong table-based reporting may require additional annotation rules and extraction targets. The thesis therefore supports claims about ML-centered scholarly IE and related scientific ERE benchmarks, but not about all forms of scientific document understanding.

Annotation and schema dependence. The thesis relies on manually defined entity and relation schemas. These schemas make extraction learnable and evaluation possible, but they also impose a specific view on the text. Distinctions such as model architecture versus concrete model, dataset versus data source, or method versus task are useful for ML literature analysis, but they can still be difficult to apply consistently. The cross-dataset experiments further show that similar labels across datasets do not necessarily encode the same annotation semantics. This means that schema alignment can support comparison, but it cannot remove the need to understand the original annotation guidelines and prediction spaces. The thesis therefore argues for schema-aware and schema-preserving evaluation, while accepting that every schema captures only one selected view of scholarly evidence.

Evidence granularity. Most extracted records in this thesis are grounded in local textual evidence, especially sentence-level contexts. This design makes extraction outputs inspectable and supports user-facing evidence views. At the same time, some ML workflow facts are distributed across several sentences, sections, tables, figures, appendices, and references. A sentence-level relation can capture many direct links, such as a model being evaluated on a dataset, but it cannot fully resolve document-level coreference, implicit comparisons, or evidence that is split across document objects. The current evidence model is therefore useful for traceable extraction, but incomplete for full reconstruction of experimental workflows. A more complete approach would need document-level inference, cross-sentence relation modeling, table and figure processing, and corpus-level artifact linking.

Model and evaluation limits. The modeling results are evaluated mainly through benchmark-style NER and RE metrics. These metrics are necessary for controlled comparison, but they do not fully capture the usefulness of extracted records in downstream exploration. A model can achieve a reasonable F1 score while still producing errors that matter strongly for users, such as merging distinct artifacts, missing important mentions, or predicting relations without enough context. The LLM-based experiments also show that flexible prompting can be useful in limited-supervision settings, but strict schema-bound extraction remains difficult under exact evaluation. Methods Miner demonstrates practical use of the extracted records, but it remains a prototype and does not yet include user correction, relation-centered exploration, robust normalization, or a full user study. The thesis therefore provides a controlled extraction and exploration basis, while leaving the validation of long-term user benefit to future work.

Together, these limitations define the boundary of the thesis contribution. The thesis does not claim to solve complete scientific document understanding or fully automatic reconstruction of all ML experiments. Its contribution is more specific: it shows how fine-grained ML artifacts and relations can be extracted from full-text scholarly publications, how these outputs can be compared across schema views, and how evidence-linked records can support inspection in an exploration system. The remaining limits point directly to the next steps: broader corpora, stronger schema-aware evaluation, document-level evidence modeling, artifact normalization, and human-in-the-loop correction.

8.4 Future Work

The results of this thesis open several directions for future work. The most important next steps concern broader evidence modeling, stronger schema-aware evaluation, interactive correction workflows, more constrained LLM-based extraction, and large-scale analysis of machine-learning research practice. Together, these directions would move the thesis contributions from controlled full-text extraction toward more complete and reusable scholarly IE systems.

Document-level and corpus-level extraction. Future work should extend sentence-level evidence with document-level coreference resolution, artifact linking, and cross-paper normalization. This is needed because ML artifacts are often introduced once, abbreviated later, referred to through generic phrases, and reused across different sections of the same paper. A document-level model could connect these mentions into paper-level artifact records and distinguish local mentions from central experimental objects. At the corpus level, extracted records should be linked across papers so that different names, aliases, versions, and spelling variants of the same dataset, model, task, or software object can be compared. Such linking would make it possible to move from mention-level extraction to stable artifact-level analysis.

Schema-aware and view-aware evaluation. Future work should further study how scientific IE resources can be compared without forcing all datasets into a single flattened schema. The cross-dataset results in this thesis show that schema alignment is useful for controlled comparison, but that simple unification can remove distinctions that matter for relation extraction. A next step is to develop evaluation protocols that preserve dataset-specific meanings while still allowing systematic comparison across resources. This could include hierarchical mappings, partial label correspondences, relation-signature-aware evaluation, and reporting methods that show which distinctions are preserved, merged, or lost. Such evaluation would better reflect the fact that datasets encode different views of scholarly evidence.

Human-in-the-loop extraction and correction. Future systems should support user correction of entity and relation records and feed these corrections back into extraction and linking models. This is important because fine-grained scholarly IE will continue to produce uncertain or context-dependent predictions. In an exploration system, users should be able to confirm, reject, merge, split, and relabel extracted artifacts and relations. These corrections could improve artifact normalization, relation extraction, and schema refinement over time. They could also provide evidence about which extraction errors matter most for real analysis tasks.

LLM-based extraction with evidence constraints. Future work should investigate LLM-based extraction methods that preserve evidence links, expose uncertainty, and remain comparable to supervised extraction models. Prompt-based extraction can adapt quickly to new schemas and limited-supervision settings, but it must be constrained so that generated records are grounded in identifiable text spans or passages. Useful directions include span-grounded generation, citation of supporting sentences,

confidence calibration, structured output validation, and direct comparison with supervised models under the same matching rules. This would make LLM-based scholarly IE more useful for evidence-oriented systems and less dependent on post-hoc interpretation of generated outputs.

Large-scale analysis of ML research practice. Future work should use extracted records to analyze dataset reuse, benchmark use, model comparison, and method diffusion across scholarly corpora. Once artifacts and relations are linked across papers, they can support empirical studies of how ML research is reported and reused. For example, extraction could reveal which datasets become standard benchmarks, which models are compared across tasks, how software and pretrained models spread across fields, and where evaluation practices change over time. Such analyses would also provide feedback for improving extraction schemas, because missing or unstable patterns in large corpora can show which artifact types and relations need better representation.

These directions extend the thesis in a consistent way. Document-level and corpus-level extraction would make the extracted records more complete. Schema-aware evaluation would make comparisons across resources more meaningful. Human-in-the-loop systems would make extraction outputs easier to correct and reuse. Evidence-constrained LLM methods would add flexibility while preserving traceability. Large-scale analysis would then use the resulting records to study how ML knowledge is reported, compared, and reused across the scholarly record.

8.5 Closing Remarks

This thesis started from the observation that understanding machine-learning publications requires more than finding isolated keywords. Models, datasets, tasks, methods, software objects, baselines, and comparisons are distributed across full texts and must be interpreted as parts of experimental workflows. The thesis has shown that these elements can be extracted as fine-grained, evidence-linked records, but also that this requires careful schema design, relation modeling, and evaluation.

The main contribution of the thesis is therefore not a single extraction model or dataset alone. It is the combined view that scholarly IE for machine-learning literature should preserve artifact distinctions, typed workflow relations, textual evidence, and schema differences at the same time. This view connects the entity-level GSAP-NER resource, the software-focused LLM extraction study, the GSAP-ERE relation resource, the schema-preserving cross-dataset evaluation, and the Methods Miner exploration system.

Taken together, these contributions provide a basis for more reliable extraction and analysis of machine-learning knowledge in scholarly publications. They show how extracted records can support evaluation, comparison, and exploration while remaining linked to the text from which they were derived. The next step is to make such records more complete across documents and corpora, more robust across schemas, and more useful in interactive systems for studying how machine-learning research is reported, compared, and reused.

Appendix

A Full List of Contributions

The following contribution statements describe my own role in the publications included in this cumulative dissertation. All included publications are co-authored works, but the form and extent of collaboration differed across papers.

For the GSAP-related contributions, especially GSAP-NER, GSAP-ERE, and Methods Miner, co-authors were involved at different stages of the research process, including discussions of the problem framing, conceptual design, data model, annotation strategy, experimental setup, implementation, analysis, and manuscript revision.

For other contributions, especially the LLM-assisted software extraction chapter and the cross-schema generalization chapter, my role covered most steps from design and implementation to experiments, analysis, and writing, while co-author input mainly concerned contextualization, methodological advice, interpretation, and revision.

The paper-specific statements below therefore make my own responsibilities explicit while also indicating where the work was shaped by collaborative discussion and co-author contributions.

A.1 GSAP-NER: A Novel Task, Corpus, and Baseline for Scholarly Entity Extraction Focused on Machine Learning Models and Datasets

Full reference. Wolfgang Otto, Matthäus Zloch, Lu Gan, Saurav Karmakar, and Stefan Dietze. *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*, Singapore, Association for Computational Linguistics, 2023, pp. 8166–8176. DOI: `https://doi.org/10.18653/v1/2023.findings-emnlp.548`.

Publication status. This manuscript was published as an archival, peer-reviewed long paper in *Findings of EMNLP 2023*.

Use in this dissertation. This manuscript forms the basis of Chapter 3. The publication text is reproduced in the layout of this dissertation and embedded through a chapter-specific context section. The chapter connects the entity-level contribution to

the later chapters on relation extraction, cross-dataset generalization, and exploration systems.

Contribution of the doctoral candidate. I was the first author of this manuscript and led the overall conception of the task, the study design, and the experimental setup.

I developed the initial idea for the fine-grained entity schema. The final entity definitions, the extent of the final label set, and the annotation decisions were developed and refined in close discussion with the co-authors.

I conceptualized and implemented the paper selection process, including the selection strategy, document preparation, and preprocessing steps needed for annotation and experimentation. The annotation guideline was developed as part of a joint process, with substantial responsibility taken by a colleague and with repeated discussion of difficult cases within the team.

My role in the annotation process was to contribute the general schema design, support the refinement of annotation decisions, participate in the joint discussions and feedback rounds with annotators, organize the technical annotation workflow, and connect the guideline decisions to the later modeling setup. The guidance of the annotation process was primarily led by me and one co-author. It involved many rounds of discussion with annotators and co-authors, especially for borderline cases, nested mentions, generic artifact mentions, and distinctions between conceptual and concrete entity types. The manual annotations were carried out by paid student annotators. I managed the annotation tool, organized the document imports, monitored annotation progress, prepared the annotated documents for later processing, and carried out the required pre- and post-processing steps. I analyzed the resulting annotation quality and contributed to the interpretation of annotation difficulties.

I designed and implemented the experimental setup for the baseline models. I designed, implemented, and conducted the baseline experiments end-to-end and was responsible for the main analysis of the results.

The writing of the manuscript was divided across authors. I coordinated the manuscript preparation, wrote the main methodological and experimental parts, and integrated the different contributions into the final version. The co-authors contributed to the conceptual refinement of the task, the discussion of entity definitions, the interpretation of annotation and modeling results, and the revision of the manuscript. I coordinated the publication process and presented the poster at EMNLP 2023.

Rights and reuse. This chapter includes the text of my own peer-reviewed ACL Anthology publication in thesis layout. The version of record is cited above. To the best of my knowledge, the inclusion of the manuscript material in this dissertation does not infringe third-party rights. No third-party material is reused beyond the material contained in the cited publication, unless explicitly indicated.

A.2 Enhancing Software-Related Information Extraction via Single-Choice Question Answering with Large Language Models

Full reference. Wolfgang Otto, Sharmila Upadhyaya, and Stefan Dietze. *Enhancing Software-Related Information Extraction via Single-Choice Question Answering with Large Language Models.* In: *Natural Scientific Language Processing and Research Knowledge Graphs.* NSLP 2024. Edited by Georg Rehm et al. Springer Nature, Cham, 2024, pp. 289–306. DOI: https://doi.org/10.1007/978-3-031-65794-8_21.

Publication status. This manuscript was published as an archival, peer-reviewed NSLP 2024 workshop paper.

Use in this dissertation. This manuscript forms the basis of Chapter 4. The publication text is reproduced in the layout of this dissertation and embedded through a chapter-specific context section. The chapter complements the supervised extraction line of the thesis by studying LLM-assisted software information extraction under limited supervision.

Contribution of the doctoral candidate. I was the first author of this manuscript and took primary responsibility for the paper from the shared-task participation to the final manuscript. I participated in the SOMD shared task and developed the methodological approach largely independently. I designed the single-choice question-answering formulation for software-related relation extraction. I also designed the retrieval-based prompting setup used in the experiments. I implemented the experimental pipeline. I ran the experiments and analyzed the results. I wrote the manuscript. The co-authors reviewed draft versions, suggested revisions to the writing, and helped improve the presentation, positioning, and final form of the paper. I presented the poster at the NSLP workshop.

Rights and reuse. This chapter includes material from my own peer-reviewed manuscript in thesis layout. The version of record is cited above. The reuse follows Springer Nature's author-reuse policy, which permits authors to reuse the Version of Record, in whole or in part, in their own thesis and to make the thesis available as required by the awarding academic institution. The reused material is included with the acknowledgement: Reproduced with permission from Springer Nature. No third-party material is reused beyond the material contained in the cited publication, unless explicitly indicated.

A.3 GSAP-ERE: Fine-Grained Scholarly Entity and Relation Extraction Focused on Machine Learning

Full reference. Wolfgang Otto, Lu Gan, Sharmila Upadhyaya, Saurav Karmakar, and Stefan Dietze. *GSAP-ERE: Fine-Grained Scholarly Entity and Relation Extraction Focused on Machine Learning.* In: *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(38), 32600–32609, 2026. DOI: https://doi.org/10.1609/aaai.v40i38.40537.

Publication status. This manuscript was published as an archival, peer-reviewed AAAI 2026 full paper.

Use in this dissertation. This manuscript forms the basis of Chapter 5. The publication text is reproduced in the layout of this dissertation and embedded through a chapter-specific context section. The chapter extends the entity-level extraction line of the dissertation to typed relations between machine-learning artifacts.

Contribution of the doctoral candidate. I was the first author of this manuscript and led the overall conception of the entity-and-relation extraction task, the dataset design, and the study design. The relation inventory, relation definitions, relation signatures, and annotation decisions were developed and refined as part of a joint conceptual process within the project team. Within this process, I contributed to structuring the relation inventory and created the seven semantic relation groups used to organize the relation types. The annotation guideline was developed collaboratively, with the second author taking primary responsibility for drafting and maintaining the guideline. The co-authors and I contributed through repeated discussions of relation definitions, difficult cases, annotation decisions, and feedback from the annotation process.

My role in the annotation process was to connect the relation-schema design, the guideline decisions, the technical annotation workflow, and the later modeling setup. The guidance of the annotation process was primarily led by the second author and me. The manual annotations were carried out by paid student annotators. I prepared the data for annotation and experimentation, managed the annotation tool, monitored annotation progress, handled annotation exports, and carried out the required pre- and post-processing steps. The second author and I analyzed the resulting annotation quality and dataset properties and interpreted the annotation difficulties together.

I was responsible for the supervised modeling approaches end-to-end. This included the design, implementation, execution, evaluation, and analysis of the supervised entity and relation extraction experiments. I contributed to the conceptual discussion of the LLM prompting baseline. The LLM prompting experiments were implemented and conducted by the third author. I contributed to the interpretation and integration of the LLM baseline results into the overall experimental discussion.

I coordinated the manuscript preparation and wrote the main parts of the paper. The co-authors contributed selected manuscript sections, conceptual refinement, result interpretation, and revisions of the paper. I coordinated the publication process, presented the paper at the conference, and presented the poster at AAAI together with the second author.

Rights and reuse. This chapter includes the text of my own peer-reviewed manuscript in thesis layout. The version of record is cited above. Permission for reuse in the electronically published version of the dissertation will be clarified with the publisher before publication, if required. To the best of my knowledge, the inclusion of the manuscript material in this submitted dissertation does not infringe third-party rights. No third-party material is reused beyond the material contained in the cited publication, unless explicitly indicated.

A.4 Generalizing Without Collapsing: Schema-Preserving Transfer Across Scientific ERE Benchmarks

Full reference. Wolfgang Otto and Stefan Dietze. *Generalizing Without Collapsing: Schema-Preserving Transfer Across Scientific ERE Benchmarks.* Submitted as long paper to EACL 2027.

Publication status. Submitted EACL 2027 long paper (Not yet accepted or published).

Use in this dissertation. This manuscript forms the basis of Chapter 6. The manuscript text is included in the layout of this dissertation and embedded through a chapter-specific context section. The chapter provides the cross-dataset generalization layer of the dissertation by comparing and aligning scientific entity and relation extraction resources.

Contribution of the doctoral candidate. I was the first author of this manuscript and initiated the central idea of the paper. I took primary responsibility for the methodological design, implementation, experiments, analysis, and manuscript preparation. I developed the alignment methodology, including the canonical schema design, the entity and relation mapping procedure, and the cross-dataset evaluation setup. The methodological framing and several design decisions were discussed intensively with the second author. The broader contextual setting of the paper was developed together with the second author in an iterative process.

I developed and implemented the schema-preserving modeling setup used for the multi-dataset experiments. I implemented the cross-dataset evaluation pipeline, prepared the datasets and mappings, ran the experiments, and analyzed the results. I developed the Dataset Explorer web tool used to inspect corpus composition, schema mappings, model predictions, and cross-dataset comparison results. I was responsible for the main interpretation of the experimental results and for connecting them to the paper's methodological argument.

I managed the writing process and wrote the manuscript. The second author contributed through intensive methodological discussion, contextualization, interpretation of the results, feedback on the argument, and revisions of the manuscript.

Rights and reuse. This chapter includes text from my own submitted manuscript in thesis layout. At the time of submission of this dissertation, the manuscript has not yet appeared as a publisher version of record. To the best of my knowledge, the inclusion of the manuscript material in this dissertation does not infringe third-party rights. No third-party material is reused beyond the material contained in the manuscript, unless explicitly indicated.

A.5 Methods Miner: Exploring Machine Learning Methods, Tasks, and Datasets in Scholarly Publications

Full reference. Wolfgang Otto, Sharmila Upadhyaya, Lu Gan, Kanishka Silva, and Stefan Dietze. *Methods Miner: Exploring Machine Learning Methods, Tasks, and Datasets in Scholarly Publications*. To be submitted.

Publication status. To be submitted as demo paper (Not yet accepted or published).

Use in this dissertation. This manuscript forms the basis of Chapter 7. The manuscript text is included in the layout of this dissertation and embedded through a chapter-specific context section. The chapter demonstrates the user-facing exploration and analysis of extracted machine-learning knowledge.

Contribution of the doctoral candidate. I was the first author of this manuscript and coordinated the overall development of the paper. The main conceptualization of the system and its intended use was developed jointly by the co-authors. I was responsible for planning the system workflow, translating the shared concept into an implementable system design, implementing major parts of the system, and integrating the processing pipeline into the final prototype. This included substantial work on PDF-to-text processing, extraction integration, data handling, storage, API components, and system-level coordination. Selected functionalities and the initial setup of selected services were developed by co-authors and integrated by me into the overall system workflow. I wrote the manuscript and managed the writing process. The co-authors contributed to system conceptualization, implementation of selected components, discussion of functionality, feedback on the demonstration scenario, and manuscript revisions.

Rights and reuse. This chapter includes text from my own submitted manuscript in thesis layout. At the time of submission of this dissertation, the manuscript has not yet appeared as a publisher version of record. To the best of my knowledge, the inclusion of the manuscript material in this dissertation does not infringe third-party rights. No third-party material is reused beyond the material contained in the manuscript, unless explicitly indicated.

B Documentation of AI-Based Tool Use

This appendix documents the use of AI-based tools during the preparation of this dissertation. These tools were used as support instruments for language, revision, and structural reflection. They did not replace the independent scientific work of the author. All scientific decisions, arguments, analyses, experiments, results, interpretations, and final formulations were checked and are the responsibility of the author. The tools used included AI-based writing and language assistance systems, in particular generative language models. They were used for language revision, wording alternatives, stylistic smoothing, shortening suggestions, structural feedback, and checks of local textual coherence. They were also used selectively to develop alternative formulations for problem statements, transitions, chapter summaries, and explanatory passages. The tools were not used as scientific sources. They were not used to generate research results, experiments, datasets, analyses, figures, tables, or conclusions without independent verification. All suggestions that were incorporated into the dissertation were checked, revised, and adapted to the context of the thesis. Scientific claims, reused material, and references to prior work are documented according to the usual rules of academic citation, independently of the use of AI-based tools. The contributions of co-authors to the publications included in this cumulative dissertation are documented separately in the corresponding sections of the thesis. In summary, AI-based tools were used as auxiliary tools for language, editing, and structural reflection. The responsibility for the content, scientific quality, and final version of the dissertation remains fully with the author.

Bibliography

Mohamed Abdalla, Jan Philip Wahle, Terry Ruas, Aurélie Névéol, Fanny Ducel, Saif Mohammad, and Karen Fort (July 2023). "The Elephant in the Room: Analyzing the Presence of Big Tech in Natural Language Processing Research". In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, pp. 13141–13160. DOI: 10.18653/v1/2023.acl-long.734.

Kritika Agrawal, Aakash Mittal, and Vikram Pudi (June 2019). "Scalable, Semi-Supervised Extraction of Structured Information from Scientific Literature". In: *Proceedings of the Workshop on Extracting Structured Knowledge from Scientific Publications*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 11–20. DOI: 10.18653/v1/W19-2602.

Beatrice Alex, Barry Haddow, and Claire Grover (June 2007). "Recognising Nested Named Entities in Biomedical Text". In: *Biological, Translational, and Clinical Language Processing*. Prague, Czech Republic: Association for Computational Linguistics, pp. 65–72. URL: https://aclanthology.org/W07-1009/.

AllenAI (2019). *scibert_scivocab_uncased (Hugging Face Model Card)*. https://huggingface.co/allenai/scibert_scivocab_uncased. Accessed: 2025-12-16.

Ron Artstein and Massimo Poesio (2008). "Survey Article: Inter-Coder Agreement for Computational Linguistics". In: *Computational Linguistics* 34.4, pp. 555–596. DOI: 10.1162/coli.07-034-R2. URL: https://aclanthology.org/J08-4004/.

Mohammad Arvan, Luís Pina, and Natalie Parde (Dec. 2022). "Reproducibility in Computational Linguistics: Is Source Code Enough?" In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 2350–2361. DOI: 10.18653/v1/2022.emnlp-main.150.

Valentin Arzt, Leonhard Hennig, Gerard De Melo, and Roland Roller (2025). "Relation Extraction or Pattern Matching? Unravelling the Generalisation of Relation Extraction Models". In: *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics. URL: https://aclanthology.org/2025.ijcnlp-long.81/.

Valentin Arzt, Abdullatif Köksal, and Iryna Gurevych (2024). "Transparency in Relation Extraction Benchmark Creation: A Case Study of TACRED and DocRED". In: *Proceedings of the 2nd Workshop on GenBench: Generalisation Benchmarking in*

NLP. Miami, Florida, USA: Association for Computational Linguistics, pp. 71–83. DOI: 10.18653/v1/2024.genbench-1.8.

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*. Vancouver, Canada: Association for Computational Linguistics, pp. 546–555. DOI: 10.18653/v1/S17-2091.

Michele Banko and Oren Etzioni (2008). "The Tradeoffs Between Open and Traditional Relation Extraction". In: *Proceedings of ACL-08: HLT*. Columbus, Ohio: Association for Computational Linguistics, pp. 28–36. URL: https://aclanthology.org/P08-1004/.

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*. Hong Kong, China: Association for Computational Linguistics, pp. 3615–3620. DOI: 10.18653/v1/D19-1371.

Emily M. Bender and Batya Friedman (2018). "Data Statements for Natural Language Processing: Toward Mitigating System Bias and Enabling Better Science". In: *Transactions of the Association for Computational Linguistics* 6, pp. 587–604. DOI: 10.1162/tacl_a_00041. URL: https://aclanthology.org/Q18-1041/.

Christine Betts, Joanna Power, and Waleed Ammar (July 2019). "GrapAL: Connecting the Dots in Scientific Literature". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Ed. by Marta R. Costa-jussà and Enrique Alfonseca. Florence, Italy: Association for Computational Linguistics, pp. 147–152. DOI: 10.18653/v1/P19-3025.

Stefano Bianchini, Moritz Müller, and Pierre Pelletier (2025). "Drivers and barriers of AI adoption and use in scientific research". In: *Technological Forecasting and Social Change* 220, p. 124303. ISSN: 0040-1625. DOI: 10.1016/j.techfore.2025.124303.

Lutz Bornmann and Rüdiger Mutz (2015). "Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references". In: *Journal of the Association for Information Science and Technology* 66.11, pp. 2215–2222. DOI: 10.1002/asi.23329.

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*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 3586–3596. DOI: 10.18653/v1/N19-1361.

Cornell University (2022). *arXiv Dataset (Kaggle), Version 123*. https://kaggle.com/datasets/Cornell-University/arxiv/versions/123. Accessed: 2025-12-16.

Jim Cowie and Wendy Lehnert (1996). "Information Extraction". In: *Communications of the ACM* 39.1, pp. 80–91. DOI: 10.1145/234173.234209.

Jennifer D'Souza and Sören Auer (2022). "Computer Science Named Entity Recognition In the Open Research Knowledge Graph". In: *From Born-Physical to Born-Virtual: Augmenting Intelligence in Digital Libraries: 24th International Conference on Asian Digital Libraries, ICADL 2022, Hanoi, Vietnam, November 30 – December 2, 2022,*

Proceedings. Hanoi, Vietnam: Springer-Verlag, pp. 35–45. ISBN: 978-3-031-21755-5. DOI: 10.1007/978-3-031-21756-2_3.

John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S. Rosen, Gerbrand Ceder, Kristin A. Persson, and Anubhav Jain (2024). "Structured information extraction from scientific text with large language models". In: *Nature Communications* 15.1418. DOI: 10.1038/s41467-024-45563-x.

Xinyu Dai, Qi Guo, Yidong Wang, Qingsong Wen, Zhen Wu, Wenjin Yao, Wei Ye, Hongbo Zhang, Meishan Zhang, Min Zhang, Shikun Zhang, Xin Zhang, and Yue Zhang (2024). "AutoSurvey: Large Language Models Can Automatically Write Surveys". In: *Advances in Neural Information Processing Systems 37*. NeurIPS 2024. Vancouver, Canada, pp. 115119–115145. DOI: 10.52202/079017-3655.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner (June 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*. Ed. by Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou. Online: Association for Computational Linguistics, pp. 4599–4610. DOI: 10.18653/v1/2021.naacl-main.365.

Danilo Dessì, Francesco Osborne, Diego Reforgiato Recupero, Davide Buscaldi, and Enrico Motta (2022). "CS-KG: A Large-Scale Knowledge Graph of Research Entities and Claims in Computer Science". In: *The Semantic Web – ISWC 2022*. Vol. 13489. Lecture Notes in Computer Science. Springer, pp. 678–696. DOI: 10.1007/978-3-031-19433-7_39.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (June 2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423/.

Caifan Du, Johanna Cohoon, Patrice Lopez, and James Howison (2021). "Softcite dataset: A dataset of software mentions in biomedical and economic research publications". In: *Journal of the Association for Information Science and Technology* 72.7, pp. 870–884. DOI: 10.1002/asi.24454.

Decheng Duan, Jitong Peng, Yingyi Zhang, and Chengzhi Zhang (Nov. 2025). "SciNLP: A Domain-Specific Benchmark for Full-Text Scientific Entity and Relation Extraction in NLP". In: *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. Ed. by Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng. Suzhou, China: Association for Computational Linguistics, pp. 14473–14486. ISBN: 979-8-89176-332-6. DOI: 10.18653/v1/2025.emnlp-main.732.

Decheng Duan, Yingyi Zhang, Jitong Peng, and Chengzhi Zhang (2026). *SciNLP: A Domain-Specific Benchmark for Full-Text Scientific Entity and Relation Extraction in NLP*. arXiv: 2509.07801v4 [cs.CL]. URL: https://arxiv.org/abs/2509.07801v4.

Michael Färber, Alexander Albers, and Felix Schüber (2021). "Identifying used methods and datasets in scientific publications". In: *Proceedings of the Workshop on Scientific Document Understanding: co-located with 35th AAAI Conference on Artificial Inteligence (AAAI 2021) ; Remote, February 9, 2021. Ed.: A. P. B. Veyseh.* Workshop on Scientific Document Understanding. SDU 2021 (Online, Feb. 19, 2021–). Vol. 2831. CEUR Workshop Proceedings. RWTH Aachen.

Marc Feger, Katarina Boland, and Stefan Dietze (July 2025). "Limited Generalizability in Argument Mining: State-Of-The-Art Models Learn Datasets, Not Arguments". In: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* Ed. by Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar. Vienna, Austria: Association for Computational Linguistics, pp. 23900–23915. ISBN: 979-8-89176-251-0. DOI: 10.18653/v1/2025.acl-long.1164.

Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach (2019). "Are we really making much progress? A worrying analysis of recent neural recommendation approaches". In: *Proceedings of the 13th ACM Conference on Recommender Systems.* RecSys '19. Copenhagen, Denmark: Association for Computing Machinery, pp. 101–109. ISBN: 9781450362436. DOI: 10.1145/3298689.3347058.

Santo Fortunato, Carl T. Bergstrom, Katy Börner, James A. Evans, Dirk Helbing, Staša Milojević, Alexander M. Petersen, Filippo Radicchi, Roberta Sinatra, Brian Uzzi, Alessandro Vespignani, Ludo Waltman, Dashun Wang, and Albert-László Barabási (2018). "Science of science". In: *Science* 359.6379, eaao0185. DOI: 10.1126/science.aao0185.

Lisheng Fu and Ralph Grishman (Nov. 2018). "A Case Study on Learning a Unified Encoder of Relations". In: *Proceedings of the First Workshop on Fact Extraction and VERification.* Ed. by Eduardo Blanco and Wei Lu. Brussels, Belgium: Association for Computational Linguistics, pp. 70–75. DOI: 10.18653/v1/W18-6126.

Kata Gábor, Davide Buscaldi, Anne-Kathrin Schumann, Behrang QasemiZadeh, Haïfa Zargayouna, and Thierry Charnois (June 2018). "SemEval-2018 Task 7: Semantic Relation Extraction and Classification in Scientific Papers". In: *Proceedings of the 12th International Workshop on Semantic Evaluation.* Ed. by Marianna Apidianaki, Saif M. Mohammad, Jonathan May, Ekaterina Shutova, Steven Bethard, and Marine Carpuat. New Orleans, Louisiana: Association for Computational Linguistics, pp. 679–688. DOI: 10.18653/v1/S18-1111.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang (2024). *Retrieval-Augmented Generation for Large Language Models: A Survey.* DOI: 10.48550/arxiv.2312.10997. arXiv: 2312.10997 [cs.CL].

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford (2021). "Datasheets for Datasets". In: *Communications of the ACM* 64.12, pp. 86–92. DOI: 10.1145/3458723.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann (2020). "Shortcut learning in deep neural networks". In: *Nature Machine Intelligence* 2, pp. 665–673. DOI: 10.1038/s42256-020-00257-z.

Genet Asefa Gesese, Zongxiong Chen, Oussama Zoubia, Fidan Limani, Kanishka Silva, Muhammad Asif Suryani, Benjamin Zapilko, Leyla Jael Castro, Ekaterina Kutafina,

Dhwani Solanki, Heike Fliegl, Sonja Schimmler, Zeyd Boukhers, and Harald Sack (2025). "A Survey on Metadata for Machine Learning Models and Datasets". In: *Proceedings of the 5th International Workshop on Scientific Knowledge: Representation, Discovery, and Assessment (Sci-K 2025)*. Vol. 4065. CEUR Workshop Proceedings, pp. 57–71. URL: https://ceur-ws.org/Vol-4065/paper05.pdf.

Behnam Ghavimi, Wolfgang Otto, and Philipp Mayr (2019). "An evaluation of the effect of reference strings and segmentation on citation matching". In: *Digital Libraries for Open Knowledge : 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019, Proceedings*. Ed. by Antoine Doucet, Antoine Isaac, Koraljka Golub, Trond Aalberg, and Adam Jatowt. Springer, Cham, pp. 365–369. DOI: 10.1007/978-3-030-30760-8_35.

Ralph Grishman (1996). "Message Understanding Conference - 6: A Brief History". In: *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*. URL: https://aclanthology.org/C96-1079/.

Honghao Gui, Lin Yuan, Hongbin Ye, Ningyu Zhang, Mengshu Sun, Lei Liang, and Huajun Chen (2024). "IEPILE: Unearthing Large-Scale Schema-Based Information Extraction Corpus". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Bangkok, Thailand: Association for Computational Linguistics, pp. 127–146. URL: https://aclanthology.org/2024.acl-short.13/.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith (July 2020). "Don't Stop Pretraining: Adapt Language Models to Domains and Tasks". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault. Online: Association for Computational Linguistics, pp. 8342–8360. DOI: 10.18653/v1/2020.acl-main.740.

Stefan Hajkowicz, Conrad Sanderson, Sarvnaz Karimi, Alexandra Bratanova, and Claire Naughtin (2023). "Artificial intelligence adoption in the physical sciences, natural sciences, life sciences, social sciences and the arts and humanities: A bibliometric analysis of research publications from 1960-2021". In: *Technology in Society* 74, p. 102260. ISSN: 0160-791X. DOI: 10.1016/j.techsoc.2023.102260.

David Harbecke, Yuxuan Chen, Leonhard Hennig, and Christoph Alt (May 2022). "Why only Micro-F1? Class Weighting of Measures for Relation Classification". In: *Proceedings of NLP Power! The First Workshop on Efficient Benchmarking in NLP*. Ed. by Tatiana Shavrina, Vladislav Mikhailov, Valentin Malykh, Ekaterina Artemova, Oleg Serikov, and Vitaly Protasov. Dublin, Ireland: Association for Computational Linguistics, pp. 32–41. DOI: 10.18653/v1/2022.nlppower-1.4.

Yichen He, Guanhua Huang, Peiyuan Feng, Yuan Lin, Yuchen Zhang, Hang Li, and Weinan E (July 2025). "PaSa: An LLM Agent for Comprehensive Academic Paper Search". In: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vienna, Austria: Association for Computational Linguistics, pp. 11663–11679. ISBN: 979-8-89176-251-0. DOI: 10.18653/v1/2025.acl-long.572. URL: https://aclanthology.org/2025.acl-long.572/.

Jenny Heddes, Pim Meerdink, Miguel Pieters, and Maarten Marx (2021). "The Automatic Detection of Dataset Names in Scientific Articles". In: *Data* 6.8, p. 84.

ISSN: 2306-5729. DOI: 10.3390/data6080084. URL: https://www.mdpi.com/2306-5729/6/8/84.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz (July 2010). "SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations Between Pairs of Nominals". In: *Proceedings of the 5th International Workshop on Semantic Evaluation*. Ed. by Katrin Erk and Carlo Strapparava. Uppsala, Sweden: Association for Computational Linguistics, pp. 33–38. URL: https://aclanthology.org/S10-1006/.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld (2011). "Knowledge-Based Weak Supervision for Information Extraction of Overlapping Relations". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon, USA: Association for Computational Linguistics, pp. 541–550. URL: https://aclanthology.org/P11-1055/.

Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly (2019). "Identification of Tasks, Datasets, Evaluation Metrics, and Numeric Scores for Scientific Leaderboards Construction". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 5203–5213. DOI: 10.18653/v1/P19-1513.

Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly (Apr. 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*. Online: Association for Computational Linguistics, pp. 707–714. DOI: 10.18653/v1/2021.eacl-main.59.

Hugging Face (2016). *Hugging Face Models (sorted by downloads)*. https://huggingface.co/models?sort=downloads. Accessed: 2022-11-13.

Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy (July 2020). "SciREX: A Challenge Dataset for Document-Level Information Extraction". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 7506–7516. DOI: 10.18653/v1/2020.acl-main.670.

Yuna Jeong and Eunhui Kim (2022). "SciDeBERTa: Learning DeBERTa for Science Technology Documents and Fine-Tuning Information Extraction Tasks". In: *IEEE Access* 10, pp. 60805–60813. DOI: 10.1109/ACCESS.2022.3180830.

Pengcheng Jiang, Jiacheng Lin, Zifeng Wang, Nigam H. Shah, and Hoifung Poon (June 2024). "GenRES: Rethinking Evaluation for Generative Relation Extraction in the Era of 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*. Ed. by Kevin Duh, Helena Gomez, and Steven Bethard. Mexico City, Mexico: Association for Computational Linguistics, pp. 2820–2837. DOI: 10.18653/v1/2024.naacl-long.155.

Wenxin Jiang, Nicholas Synovic, Matt Hyatt, Taylor R. Schorlemmer, Rohan Sethi, Yung-Hsiang Lu, George K. Thiruvathukal, and James C. Davis (2023). "An Empirical Study of Pre-Trained Model Reuse in the Hugging Face Deep Learning Model Registry". In: *Proceedings of the 45th International Conference on Software*

Engineering. ICSE '23. Melbourne, Victoria, Australia: IEEE Press, pp. 2463–2475. ISBN: 9781665457019. DOI: 10.1109/ICSE48619.2023.00206.

Michael I. Jordan and Tom M. Mitchell (2015). "Machine learning: Trends, perspectives, and prospects". In: *Science* 349.6245, pp. 255–260. DOI: 10.1126/science.aaa8415.

Salomon Kabongo, Jennifer D'Souza, and Sören Auer (2021). "Automated Mining Of Leaderboards For Empirical AI Research". In: *Towards Open and Trustworthy Digital Societies: 23rd International Conference on Asia-Pacific Digital Libraries, ICADL 2021, Virtual Event, December 1–3, 2021, Proceedings*. Berlin, Heidelberg: Springer-Verlag, pp. 453–470. ISBN: 978-3-030-91668-8. DOI: 10.1007/978-3-030-91669-5_35.

Sayash Kapoor, Emily M. Cantrell, Kenny Peng, Thanh Hien Pham, Christopher A. Bail, Odd Erik Gundersen, Jake M. Hofman, Jessica Hullman, Michael A. Lones, Momin M. Malik, Priyanka Nanayakkara, Russell A. Poldrack, Inioluwa Deborah Raji, Michael Roberts, Matthew J. Salganik, Marta Serra-Garcia, Brandon M. Stewart, Gilles Vandewiele, and Arvind Narayanan (2024). "REFORMS: Consensus-based Recommendations for Machine-learning-based Science". In: *Science Advances* 10.18, eadk3452. DOI: 10.1126/sciadv.adk3452. eprint: https://www.science.org/doi/pdf/10.1126/sciadv.adk3452. URL: https://www.science.org/doi/abs/10.1126/sciadv.adk3452.

Marcin Kardas, Piotr Czapla, Pontus Stenetorp, Sebastian Ruder, Sebastian Riedel, Ross Taylor, and Robert Stojnic (Nov. 2020). "AxCell: Automatic Extraction of Results from Machine Learning Papers". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 8580–8594. DOI: 10.18653/v1/2020.emnlp-main.692.

Tetsu Kasanishi, Masaru Isonuma, Junichiro Mori, and Ichiro Sakata (July 2023). "SciReviewGen: A Large-scale Dataset for Automatic Literature Review Generation". In: *Findings of the Association for Computational Linguistics: ACL 2023*. Toronto, Canada: Association for Computational Linguistics, pp. 6695–6715.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych (Aug. 2018). "The INCEpTION Platform: Machine-Assisted and Knowledge-Oriented Interactive Annotation". In: *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*. Ed. by Dongyan Zhao. Santa Fe, New Mexico: Association for Computational Linguistics, pp. 5–9. URL: https://aclanthology.org/C18-2002/.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton (2015). "Deep learning". In: *Nature* 521.7553, pp. 436–444. DOI: 10.1038/nature14539.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela (2020). "Retrieval-augmented generation for knowledge-intensive NLP tasks". In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS '20. Vancouver, BC, Canada: Curran Associates Inc. ISBN: 9781713829546. DOI: 10.5555/3495724.3496517.

Qi Li and Heng Ji (2014). "Incremental Joint Extraction of Entity Mentions and Relations". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers*. Baltimore, Maryland: Association for Computational Linguistics, pp. 402–412. DOI: 10.3115/v1/P14-1038.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li (July 2019). "Entity-Relation Extraction as Multi-Turn Question Answering". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Ed. by Anna Korhonen, David Traum, and Lluís Màrquez. Florence, Italy: Association for Computational Linguistics, pp. 1340–1350. DOI: 10.18653/v1/P19-1129.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig (Jan. 2023). "Pre-Train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing". In: *ACM Comput. Surv.* 55.9. ISSN: 0360-0300. DOI: 10.1145/3560815.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer (2020). "Multilingual Denoising Pre-training for Neural Machine Translation". In: *Transactions of the Association for Computational Linguistics*. Vol. 8, pp. 726–742.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. arXiv: 1907.11692 [cs.CL].

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S. Weld (2020). "S2ORC: The Semantic Scholar Open Research Corpus". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 4969–4983. DOI: 10.18653/v1/2020.acl-main.447.

Patrice Lopez (2008). *GROBID (GitHub Repository)*. https://github.com/kermitt2/grobid. Accessed: 2025-12-16.

Patrice Lopez (2009). "GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction for Scholarship Publications". In: *Research and Advanced Technology for Digital Libraries (ECDL 2009)*. DOI: 10.1007/978-3-642-04346-8_62.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu (May 2022). "Unified Structure Generation for Universal Information Extraction". In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Volume 1, Long Papers*. Ed. by Smaranda Muresan, Preslav Nakov, and Aline Villavicencio. Dublin, Ireland: Association for Computational Linguistics, pp. 5755–5772. DOI: 10.18653/v1/2022.acl-long.395.

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*. Brussels, Belgium: Association for Computational Linguistics, pp. 3219–3232. DOI: 10.18653/v1/D18-1360.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi (June 2019). "A general framework for information extraction using dynamic span graphs". 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)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 3036–3046. DOI: 10.18653/v1/N19-1308.

Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun (Dec. 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*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 10572–10601. DOI: 10.18653/v1/2023.findings-emnlp.710.

Aman Madaan, Katherine Hermann, and Amir Yazdanbakhsh (Dec. 2023). "What Makes Chain-of-Thought Prompting Effective? A Counterfactual Study". In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 1448–1535. DOI: 10.18653/v1/2023.findings-emnlp.101.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky (2009). "Distant Supervision for Relation Extraction without Labeled Data". In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Suntec, Singapore: Association for Computational Linguistics, pp. 1003–1011. DOI: 10.3115/1690219.1690287.

Makoto Miwa and Mohit Bansal (2016). "End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers*. Berlin, Germany: Association for Computational Linguistics, pp. 1105–1116. DOI: 10.18653/v1/P16-1105.

Ishani Mondal, Yufang Hou, and Charles Jochim (Aug. 2021). "End-to-End Construction of NLP Knowledge Graph". In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics, pp. 1885–1895. DOI: 10.18653/v1/2021.findings-acl.165.

Zara Nasar, Syed Waqar Jaffry, and Muhammad Kamran Malik (2018). "Information Extraction from Scientific Articles: A Survey". In: *Scientometrics* 117.3, pp. 1931–1990. ISSN: 0138-9130. DOI: 10.1007/s11192-018-2921-5.

Natural Language Processing Lab at Tsinghua University (2022). *PL-Marker (GitHub Repository)*. https://github.com/thunlp/PL-Marker. Accessed: 2025-12-16.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh (2018). "A Survey on Open Information Extraction". In: *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, pp. 3866–3878. URL: https://aclanthology.org/C18-1326/.

Allard Oelen, Jennifer D'Souza, Markus Stocker, and Sören Auer (2024). "Creating and Validating a Scholarly Knowledge Graph Using Natural Language Processing and Microtask Crowdsourcing". In: *International Journal on Digital Libraries* 25, pp. 365–380. DOI: 10.1007/s00799-023-00360-7.

Ollama (2024). *Ollama: Run Large Language Models Locally*. https://ollama.com/. Accessed: 2025-12-16.

Ollama (2025a). *llama3.1:70b-instruct-q4_K_M (Ollama Library Entry)*. https://ollama.com/library/llama3.1:70b-instruct-q4_K_M. Accessed: 2025-12-16.

Ollama (2025b). *qwen2.5:32b (Ollama Library Entry)*. https://ollama.com/library/qwen2.5:32b. Accessed: 2025-12-16.

Wolfgang Otto (2018). "Team GESIS Cologne: An all in all sentence-based approach for FEVER". In: *Proceedings of the First Workshop on Fact Extraction and VERification*

(FEVER). Ed. by James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. Association for Computational Linguistics, pp. 145–149. DOI: 10.18653/v1/W18-5524.

Wolfgang Otto, Lu Gan, Sharmila Upadhyaya, Saurav Karmakar, and Stefan Dietze (2025). *GSAP-ERE: Fine-Grained Scholarly Entity and Relation Extraction Focused on Machine Learning*. arXiv: 2511.09411 [cs.CL]. URL: https://arxiv.org/abs/2511.09411.

Wolfgang Otto, Lu Gan, Sharmila Upadhyaya, Saurav Karmakar, and Stefan Dietze (Mar. 2026). "GSAP-ERE: Fine-Grained Scholarly Entity and Relation Extraction Focused on Machine Learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 40.38, pp. 32600–32609. DOI: 10.1609/aaai.v40i38.40537.

Wolfgang Otto, Behnam Ghavimi, Philipp Mayr, Rajesh Piryani, and Vivek Kumar Singh (2019). "Highly cited references in PLOS ONE and their in-text usage over time". In: *Proceedings of the 17th International Conference on Scientometrics & Informetrics (ISSI 2019)*. Ed. by Giuseppe Catalano, Cinzia Daraio, Martina Gregori, Henk F. Moed, and Giancarlo Ruocco. Rome: Edizioni Efesto, pp. 1531–1536. URL: https://www.issi-society.org/proceedings/issi_2019/ISSI%202019%20-%20Proceedings%20VOLUME%20II.pdf.

Wolfgang Otto, Sharmila Upadhyaya, and Stefan Dietze (2024). "Enhancing Software-Related Information Extraction via Single-Choice Question Answering with Large Language Models". In: *Natural Scientific Language Processing and Research Knowledge Graphs. NSLP 2024*. Ed. by Georg Rehm, Stefan Dietze, Sonja Schimmler, and Frank Krüger. Springer Nature, Cham, pp. 289–306. DOI: 10.1007/978-3-031-65794-8_21.

Wolfgang Otto, Andrea Zielinski, Behnam Ghavimi, Dimitar Dimitrov, Narges Tavakolpoursaleh, Karam Abdulahhad, Katarina Boland, and Stefan Dietze (2020). "Knowledge extraction from scholarly publications. The GESIS contribution to the rich context competition". In: *Rich search and discovery for research datasets*. Ed. by Julia Lane, Ian Mulvany, and Paco Nathan. Sage, Los Angeles u.a., pp. 107–127. URL: https://study.sagepub.com/richcontext.

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*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Association for Computational Linguistics, Singapore, pp. 8166–8176. URL: https://aclanthology.org/2023.findings-emnlp.548.

Huitong Pan, Qi Zhang, Cornelia Caragea, Eduard Dragut, and Longin Jan Latecki (May 2024). "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)*. Ed. by Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue. Torino, Italia: ELRA and ICCL, pp. 14407–14417. URL: https://aclanthology.org/2024.lrec-main.1256/.

Huitong Pan, Qi Zhang, Eduard Dragut, Cornelia Caragea, and Longin Jan Latecki (2023). "DMDD: A Large-Scale Dataset for Dataset Mentions Detection". In: *Transactions of the Association for Computational Linguistics* 11, pp. 1132–1146. DOI: 10.1162/tacl_a_00592.

Yannis Papanikolaou, Marlene Staib, Justin Grace, and Francine Bennett (2022). "Slot Filling for Biomedical Information Extraction". In: *Proceedings of the 21st Workshop on Biomedical Language Processing*. Dublin, Ireland: Association for Computational Linguistics, pp. 82–90. URL: https://aclanthology.org/2022.bionlp-1.7/.

Dong Pham, Xanh Ho, Quang Thuy Ha, and Akiko Aizawa (Dec. 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*. Ed. by Chu-Ren Huang, Yasunari Harada, Jong-Bok Kim, Si Chen, Yu-Yin Hsu, Emmanuele Chersoni, Pranav A, Winnie Huiheng Zeng, Bo Peng, Yuxi Li, and Junlin Li. Hong Kong, China: Association for Computational Linguistics, pp. 243–256. URL: https://aclanthology.org/2023.paclic-1.25/.

Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Lariviere, Alina Beygelzimer, Florence d'Alche-Buc, Emily Fox, and Hugo Larochelle (2021). "Improving Reproducibility in Machine Learning Research(A Report from the NeurIPS 2019 Reproducibility Program)". In: *Journal of Machine Learning Research* 22.164, pp. 1–20. URL: http://jmlr.org/papers/v22/20-303.html.

Rajesh Piryani, Wolfgang Otto, Philipp Mayr, and Vivek Kumar Singh (2019). "Analysing author name mentions in citation contexts of highly cited publications". In: *Proceedings of the 4th Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL 2019)*. Ed. by Muthu Kumar Chandrasekaran and Philipp Mayr. RWTH, Aachen, pp. 145–152. URL: http://ceur-ws.org/Vol-2414/paper16.pdf.

Animesh Prasad, Chenglei Si, and Min-Yen Kan (June 2019). "Dataset Mention Extraction and Classification". In: *Proceedings of the Workshop on Extracting Structured Knowledge from Scientific Publications*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 31–36. DOI: 10.18653/v1/W19-2604.

Behrang QasemiZadeh and Anne-Kathrin Schumann (May 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)*. Portorož, Slovenia: European Language Resources Association (ELRA), pp. 1862–1868. URL: https://aclanthology.org/L16-1294.

Edward Raff (2019). "A Step Toward Quantifying Independently Reproducible Machine Learning Research". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/c429429bf1f2af051f2021dc92a8ebea-Paper.pdf.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu (2020). "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". In: *Journal of Machine Learning Research* 21, pp. 1–67.

Nils Reimers and Iryna Gurevych (Nov. 2019). "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". 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)*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Hong Kong, China: Association for Computational Linguistics, pp. 3982–3992. DOI: 10.18653/v1/D19-1410.

Alan Ritter, Luke Zettlemoyer, Mausam, and Oren Etzioni (2013). "Modeling Missing Data in Distant Supervision for Information Extraction". In: *Transactions of the Association for Computational Linguistics* 1, pp. 367–378. DOI: 10.1162/tacl_a_00234. URL: https://aclanthology.org/Q13-1030/.

Shaurya Rohatgi, Yanxia Qin, Benjamin Aw, Niranjana Unnithan, and Min-Yen Kan (2023). "The ACL OCL Corpus: Advancing Open Science in Computational Linguistics". In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Singapore: Association for Computational Linguistics, pp. 10547–10561. DOI: 10.18653/v1/2023.emnlp-main.640.

Ayahito Saji and Shigeki Matsubara (2022). "Extracting Information about Research Resources from Scholarly Papers". In: *From Born-Physical to Born-Virtual: Augmenting Intelligence in Digital Libraries*. Ed. by Yuen-Hsien Tseng, Marie Katsurai, and Hoa N. Nguyen. Cham: Springer International Publishing, pp. 440–448. ISBN: 978-3-031-21756-2. DOI: 10.1007/978-3-031-21756-2_35.

David Schindler, Felix Bensmann, Stefan Dietze, and Frank Krüger (2021). "SoMeSci: A 5 Star Open Data Gold Standard Knowledge Graph of Software Mentions in Scientific Articles". In: *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*. New York, NY, USA: Association for Computing Machinery, pp. 4574–4583. DOI: 10.1145/3459637.3482017.

David Schindler, Felix Bensmann, Stefan Dietze, and Frank Krüger (2022). "The Role of Software in Science: A Knowledge Graph-Based Analysis of Software Mentions in PubMed Central". In: *PeerJ Computer Science* 8, e835. DOI: 10.7717/peerj-cs.835.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo (June 2013). "SemEval-2013 Task 9: Extraction of Drug-Drug Interactions from Biomedical Texts". In: *Second Joint Conference on Lexical and Computational Semantics and Seventh International Workshop on Semantic Evaluation*. Ed. by Suresh Manandhar and Deniz Yuret. Atlanta, Georgia, USA: Association for Computational Linguistics, pp. 341–350. URL: https://aclanthology.org/S13-2056/.

Sentence-transformers (2025). *multi-qa-mpnet-base-cos-v1 (Hugging Face Model Card)*. https://huggingface.co/sentence-transformers/multi-qa-mpnet-base-cos-v1. Accessed: 2025-12-16.

Mahsa Shamsabadi, Jennifer D'Souza, and Sören Auer (Mar. 2024). "Large Language Models for Scientific Information Extraction: An Empirical Study for Virology". In: *Findings of the Association for Computational Linguistics: EACL 2024*. Ed. by Yvette Graham and Matthew Purver. St. Julian's, Malta: Association for Computational Linguistics, pp. 392–408. DOI: 10.18653/v1/2024.findings-eacl.26.

Étienne Simon, Helene Bøsei Olsen, Huiling You, Samia Touileb, Lilja Øvrelid, and Erik Velldal (2024). "Generative Approaches to Event Extraction: Survey and Outlook". In: *Proceedings of the Workshop on the Future of Event Detection*. Miami, Florida, USA: Association for Computational Linguistics, pp. 73–86. URL: https://aclanthology.org/2024.futured-1.7/.

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman (Dec. 2023). "SciRepEval: A Multi-Format Benchmark for Scientific Document Representations". In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Ed. by Houda Bouamor, Juan Pino, and Kalika

Bali. Singapore: Association for Computational Linguistics, pp. 5548–5566. DOI: 10.18653/v1/2023.emnlp-main.338.

Mayank Singh, Barnopriyo Barua, Priyank Palod, Manvi Garg, Sidhartha Satapathy, Samuel Bushi, Kumar Ayush, Krishna Sai Rohith, Tulasi Gamidi, Pawan Goyal, and Animesh Mukherjee (2016). "OCR++: A Robust Framework For Information Extraction from Scholarly Articles". In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan: The COLING 2016 Organizing Committee, pp. 3390–3400. URL: https://aclanthology.org/C16-1320/.

Michal Štefánik, Timothee Mickus, Michal Spiegel, Marek Kadlčík, and Josef Kuchař (Nov. 2025). "Can Out-of-Distribution Evaluations Uncover Reliance on Prediction Shortcuts? A Case Study in Question Answering". In: *Findings of the Association for Computational Linguistics: EMNLP 2025*. Ed. by Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng. Suzhou, China: Association for Computational Linguistics, pp. 22628–22635. ISBN: 979-8-89176-335-7. DOI: 10.18653/v1/2025.findings-emnlp.1232.

George Stoica, Emmanouil Antonios Platanios, and Barnabás Póczos (2021). "Re-TACRED: Addressing Shortcomings of the TACRED Dataset". In: *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*. AAAI Press, pp. 13843–13850. URL: https://ojs.aaai.org/index.php/AAAI/article/view/17631.

Jana Straková and Milan Straka (July 2025). "NameTag 3: A Tool and a Service for Multilingual/Multitagset NER". In: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*. Ed. by Pushkar Mishra, Smaranda Muresan, and Tao Yu. Vienna, Austria: Association for Computational Linguistics, pp. 31–39. ISBN: 979-8-89176-253-4. DOI: 10.18653/v1/2025.acl-demo.4.

Erik F. Tjong Kim Sang and Fien De Meulder (2003). "Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition". In: *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pp. 142–147. URL: https://aclanthology.org/W03-0419/.

Dominika Tkaczyk, Paweł Szostek, Mateusz Fedoryszak, Piotr Jan Dendek, and Łukasz Bolikowski (2015). "CERMINE: automatic extraction of structured metadata from scientific literature". In: *International Journal on Document Analysis and Recognition* 18.4, pp. 317–335. DOI: 10.1007/s10032-015-0249-8.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample (2023). *LLaMA: Open and Efficient Foundation Language Models*. DOI: 10.48550/arXiv.2302.13971. arXiv: 2302.13971 [cs.CL].

Sharmila Upadhyaya, Wolfgang Otto, Frank Krüger, and Stefan Dietze (2025). "SOMD 2025: A Challenging Shared Task for Software Related Information Extraction". In: *Proceedings of the Fifth Workshop on Scholarly Document Processing (SDP 2025)*, pp. 137–145.

Vijay Viswanathan, Graham Neubig, and Pengfei Liu (Aug. 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). Online: Association for Computational Linguistics, pp. 719–731. DOI: 10.18653/v1/2021.acl-long.59.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi (Nov. 2020). "Fact or Fiction: Verifying Scientific Claims". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 7534–7550. DOI: 10.18653/v1/2020.emnlp-main.609.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi (Nov. 2019). "Entity, Relation, and Event Extraction with Contextualized Span Representations". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Hong Kong, China: Association for Computational Linguistics, pp. 5784–5789. DOI: 10.18653/v1/D19-1585.

Somin Wadhwa, Silvio Amir, and Byron Wallace (July 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)*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, pp. 15566–15589. DOI: 10.18653/v1/2023.acl-long.868.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi (Dec. 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*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 3534–3547. DOI: 10.18653/v1/2023.emnlp-main.214.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo (Apr. 2025). "GPT-NER: Named Entity Recognition via Large Language Models". In: *Findings of the Association for Computational Linguistics: NAACL 2025*. Ed. by Luis Chiruzzo, Alan Ritter, and Lu Wang. Albuquerque, New Mexico: Association for Computational Linguistics, pp. 4257–4275. ISBN: 979-8-89176-195-7. DOI: 10.18653/v1/2025.findings-naacl.239.

Xingbo Wang, Samantha L. Huey, Rui Sheng, Saurabh Mehta, and Fei Wang (2025). "SciDaSynth: Interactive Structured Data Extraction from Scientific Literature with Large Language Model". In: *Campbell Systematic Reviews* 21, pp. 1–16. DOI: 10.1002/cl2.70073.

Joseph Wonsil, Jack Sullivan, Margo Seltzer, and Adam Pocock (2023). "Integrated Reproducibility with Self-describing Machine Learning Models". In: *Proceedings of the 2023 ACM Conference on Reproducibility and Replicability*. ACM REP '23. Santa Cruz, CA, USA: Association for Computing Machinery, pp. 1–14. ISBN: 9798400701764. DOI: 10.1145/3589806.3600039.

Tingyu Xie, Qi Li, Yan Zhang, Zuozhu Liu, and Hongwei Wang (June 2024). "Self-Improving for Zero-Shot Named Entity Recognition with Large Language Models". In: *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*. Ed. by Kevin Duh, Helena Gomez, and Steven Bethard. Mexico

City, Mexico: Association for Computational Linguistics, pp. 583–593. DOI: 10.18653/v1/2024.naacl-short.49.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen (2024). "Large Language Models for Generative Information Extraction: A Survey". In: *Frontiers of Computer Science* 18.6, p. 186357. ISSN: 2095-2236. DOI: 10.1007/s11704-024-40555-y.

Zhaohui Yan, Songlin Yang, Wei Liu, and Kewei Tu (Dec. 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*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 7512–7526. DOI: 10.18653/v1/2023.emnlp-main.467.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu (2025). *Qwen2.5 Technical Report*. arXiv: 2412.15115 [cs.CL]. URL: https://arxiv.org/abs/2412.15115.

Yang Yang, Zhilei Wu, Yuexiang Yang, Shuangshuang Lian, Fengjie Guo, and Zhiwei Wang (2022). "A Survey of Information Extraction Based on Deep Learning". In: *Applied Sciences* 12.19, p. 9691. DOI: 10.3390/app12199691.

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun (May 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)*. Ed. by Smaranda Muresan, Preslav Nakov, and Aline Villavicencio. Dublin, Ireland: Association for Computational Linguistics, pp. 4904–4917. DOI: 10.18653/v1/2022.acl-long.337.

Hongbin Ye, Ningyu Zhang, Hui Chen, and Huajun Chen (Dec. 2022). "Generative Knowledge Graph Construction: A Review". In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 1–17. URL: https://aclanthology.org/2022.emnlp-main.1.

Raquib Bin Yousuf, Subhodip Biswas, Kulendra Kumar Kaushal, James Dunham, Rebecca Gelles, Sathappan Muthiah, Nathan Self, Patrick Butler, and Naren Ramakrishnan (2022). "Lessons from Deep Learning applied to Scholarly Information Extraction: What Works, What Doesn't, and Future Directions". In: *arXiv preprint arXiv:2207.04029*. URL: https://arxiv.org/abs/2207.04029.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella (2002). "Kernel Methods for Relation Extraction". In: *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pp. 71–78. DOI: 10.3115/1118693.1118703. URL: https://aclanthology.org/W02-1010/.

Bowen Zhang and Harold Soh (Nov. 2024). "Extract, Define, Canonicalize: An LLM-based Framework for Knowledge Graph Construction". In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA: Association

for Computational Linguistics, pp. 9827–9847. DOI: 10.18653/v1/2024.emnlp-main.548.

Qi Zhang, Zhijia Chen, Huitong Pan, Cornelia Caragea, Longin Jan Latecki, and Eduard Dragut (Nov. 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*. Ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Association for Computational Linguistics, pp. 13083–13100. DOI: 10.18653/v1/2024.emnlp-main.726.

Xingjian Zhang, Yutong Xie, Jin Huang, Jinge Ma, Zhaoying Pan, Qijia Liu, Ziyang Xiong, Tolga Ergen, Dongsub Shim, Honglak Lee, and Qiaozhu Mei (Apr. 2025). "MASSW: A New Dataset and Benchmark Tasks for AI-Assisted Scientific Workflows". In: *Findings of the Association for Computational Linguistics: NAACL 2025*. Ed. by Luis Chiruzzo, Alan Ritter, and Lu Wang. Albuquerque, New Mexico: Association for Computational Linguistics, pp. 2373–2394. ISBN: 979-8-89176-195-7. DOI: 10.18653/v1/2025.findings-naacl.127. URL: https://aclanthology.org/2025.findings-naacl.127/.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning (Sept. 2017). "Position-aware Attention and Supervised Data Improve Slot Filling". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Ed. by Martha Palmer, Rebecca Hwa, and Sebastian Riedel. Copenhagen, Denmark: Association for Computational Linguistics, pp. 35–45. DOI: 10.18653/v1/D17-1004.

Zikang Zhang, Wangjie You, Tianci Wu, Xinrui Wang, Juntao Li, and Min Zhang (2025). "A Survey of Generative Information Extraction". In: *Proceedings of the 31st International Conference on Computational Linguistics*. Abu Dhabi, UAE: Association for Computational Linguistics, pp. 4840–4870. URL: https://aclanthology.org/2025.coling-main.324/.

Zhaohui Yan (2023). *HGERE (GitHub Repository)*. https://github.com/yanzhh/HGERE. Accessed: 2025-12-16.

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 (NAACL)*, pp. 50–61. DOI: 10.18653/v1/2021.naacl-main.5.