INFORMATION EXTRACTION FROM SCIENTIFIC LITERATURE

A Dissertation
Submitted to
the Temple University Graduate Board

In Partial Fulfillment
of the Requirements for the Degree
DOCTOR OF PHILOSOPHY

---

by
Huitong Pan
May 2025

Examining Committee Members:

Longin Jan Latecki, Advisory Chair, Computer and Information Sciences
Eduard C. Dragut, Computer and Information Sciences
Hongchang Gao, Computer and Information Sciences
Cornelia Caragea, External Reader, University of Illinois Chicago

ABSTRACT

The exponential growth of scientific literature, with millions of new articles published annually, has created an unsustainable discovery bottleneck across research communities. Manual extraction of critical information—including methodologies, datasets, and domain-specific terminologies—now consumes a substantial proportion of researchers' literature review time, particularly impacting time-sensitive fields like climate science and biomedical research where delayed insights hinder urgent policy decisions or therapeutic developments. Automated information extraction systems have transitioned from supplemental tools to essential infrastructure, addressing three critical imperatives: preserving collective understanding through cross-publication discovery linking, enabling real-time knowledge synthesis in rapidly evolving domains, and democratizing access to specialized findings via structured knowledge representation. Without robust frameworks, the scientific community risks perpetuating redundant investigations, overlooking critical interdisciplinary connections, and failing to transform publication volume into actionable insight networks.

Current information extraction paradigms face four fundamental technical challenges rooted in scientific communication's unique characteristics. First, terminological instability arises from continuous conceptual evolution, where emerging constructs like "attribution-based climate models" and "GPT-4.5" outpace standardized taxonomies, generating persistent errors in entity disambiguation. Second, structural heterogeneity manifests through hundreds of distinct

methodological description formats observed even within focused disciplines like materials science, complicating pattern generalization. Third, contextual dependency demands adaptive interpretation of concepts such as "deep learning," whose technical meanings diverge fundamentally between protein folding architectures and geospatial mapping applications. Fourth, the scalability–accuracy tradeoff forces untenable compromises between precision (evidenced by frequent LLM hallucinations) and coverage (marked by traditional NLP's oversight of domain-specific abbreviations). These technical barriers compound with systemic data limitations—existing corpora cover only a small fraction of specialized domains while exhibiting annotation inconsistencies that undermine model reliability. Emerging paradigms in hierarchical relationship modeling hint at potential resolutions through hybrid neural-symbolic architectures.

This research advances scientific information extraction through three interconnected contributions: the development of domain-annotated corpora spanning climate science and computer science; systematic evaluation of machine learning architectures across extraction tasks and disciplinary contexts; and demonstrated pathways for transforming extracted entities into evolvable knowledge graphs. By creating structured repositories that capture methodological lineages, dataset dependencies, and conceptual evolution patterns, our work provides researchers with interoperable frameworks for mapping relationships across fragmented scientific domains. The resulting infrastructure enables both precision-focused analysis within specialized fields and cross-domain knowledge discovery, offering scalable solutions to organize literature at scale while preserving disciplinary

nuance. These contributions collectively address the dual challenges of maintaining taxonomic rigor and enabling adaptive knowledge synthesis in modern scientific communication ecosystems.

To God, my family, friends, and advisors, I express my deepest gratitude for their guidance, support, and encouragement throughout this journey.

ACKNOWLEDGEMENTS

First and foremost, I give profound thanks to God for granting me the strength, wisdom, and perseverance to complete this doctoral journey. "I can do all things through Christ who strengthens me" (Philippians 4:13) — this truth anchored me when human effort faltered, and His wisdom guided my steps when the path grew unclear. To my beloved family - my parents, siblings, and extended relatives - your unwavering love and patient encouragement sustained me through moments of doubt, and for this I am eternally grateful.

I owe deepest gratitude to my advisor, Prof. Longin Jan Latecki, whose steadfast guidance shaped this research from conception to completion. Your inspirational mentorship during our most challenging discussions renewed my confidence to push intellectual boundaries. To my co-advisor, Prof. Eduard C. Dragut, thank you for your exacting standards in methodology design and those transformative late-night paper reviews that fundamentally improved this work. Together, your complementary insights created the ideal environment for scholarly growth.

I extend sincere appreciation to my dissertation committee members, Prof. Hongchang Gao and Prof. Cornelia Caragea, for their rigorous engagement and expert guidance.

To my pillars of support in the lab and beyond - Qi Zhang, Sidra Hanif, Zhiqian Ye, Xixi Shen, Yuying Chen, Merlyn Wu, and Bochen Huang - thank you for the collaborative efforts, the shared laughter that dissolved stress, and the spiritual camaraderie that reminded me why this journey matters.

To all who contributed to this milestone, named and unnamed, you have my enduring gratitude. The completion of this dissertation is a reflection of the collective support, guidance, and encouragement I received from those around me.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The scale of modern scientific publishing has reached critical thresholds, with researchers submitting over 7 million papers annually. In biomedicine alone, PubMed indexes 58K new abstracts relating to 'covid' last year—a volume that would take a researcher more than a year to read nonstop at 1 minute per paper. This deluge creates discovery bottlenecks: critical findings on vaccine efficacy, for example, took 6 months longer to synthesize during the COVID-19 pandemic due to fragmented literature. Compounding this, interdisciplinary fields like climate science now require synthesizing insights from diverse domains (atmospheric chemistry, AI-driven modeling, policy studies) just to validate a single hypothesis. Without automated tools, researchers risk missing vital connections—many clinical trials proceed without conducting thorough systematic reviews of existing evidence. This oversight can result in unnecessary duplication and exposure of participants to avoidable risks [1].

Automatic information extraction (IE) systems have emerged as essential infrastructure to address these challenges, encompassing three core tasks: named entity recognition, relationship detection, and entity linking as shown in Figure 1.1. Their transformative potential is evidenced by large-scale implementations such as AllenAI's Semantic Scholar for scientific literature structuring and the National Institutes of Health's LitCOVID knowledge graph [2], which codified COVID-19 research into actionable biomedical relationships. These systems aim to convert unstructured text into semantically rich, queryable knowledge networks.

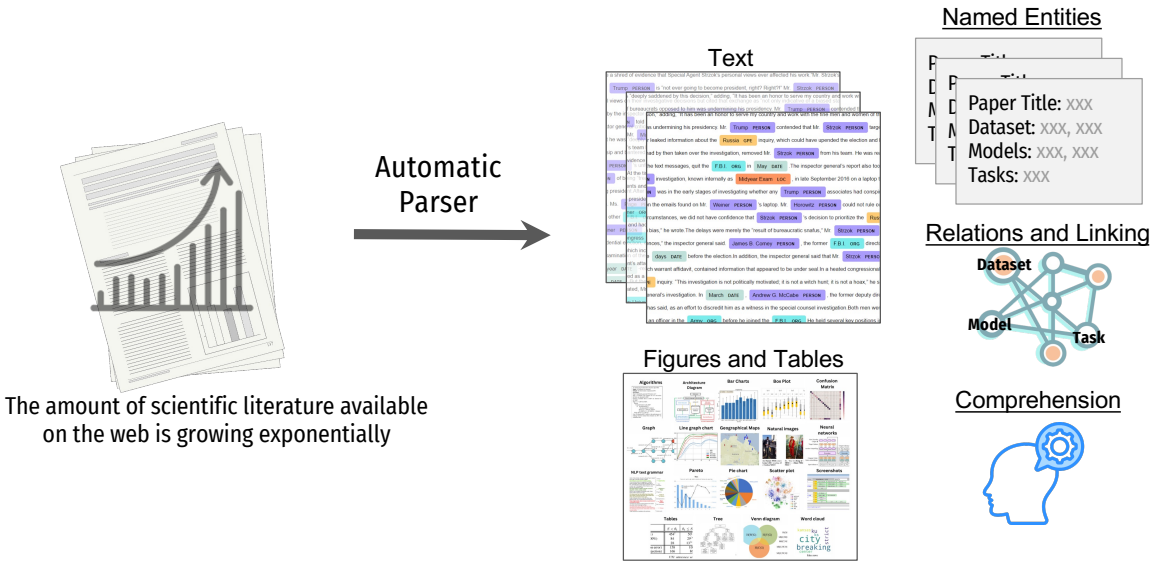


Figure 1.1: Background and motivation of automatic information extraction systems on scientific literature

Despite these advances, domain-specific complexities persist. Terminological instability is exemplified by the evolving semantics of "awe"—initially denoting emotional reverence in psychology, later redefined as algorithmic wonder in human-AI interaction studies [3]. Structural heterogeneity manifests as hundreds of methodological description formats within singular disciplines; materials scientists, for instance, document identical thermal processes using lexically divergent terms like "calcined," "fired," and "heated" [4], complicating pattern generalization. Cross-modal ambiguity further obscures meaning, as seen in "resilience," which signifies species recovery rates in ecological contexts but denotes system robustness thresholds in cybersecurity frameworks. Such variability demands adaptive solutions to align extracted knowledge with disciplinary conventions.

This thesis advances scientific information extraction through three methodological pillars:

1. Domain-Specific Corpora Development Developed four benchmark resources for scientific IE: *DMDD* [5]: Dataset-method discourse corpus with 449,000 weak labels and 13,000 expert annotations across computer science literature. *SciDMT* [6]: Dataset-method-task triples from 48,000 papers, revealing implicit research workflows. *FlowLearn* [7]: 3,858 annotated scientific flowcharts supplemented with 10,000 synthetic diagrams for visual-language pretraining. *ClimateIE*: 500 climate publications annotated via hybrid human-AI pipeline, aligned with IPCC taxonomies through entity linking.

2. Cross-Model Evaluation Systematically evaluated multiple state-of-the-art models across five IE tasks using an overlapping set of architectures:

- *Entity Detection*: Cross-paradigm comparison of LLMs (GPT-4, Llama) versus domain-specialized architectures (SciBERT)
- *Relationship Extraction*: Systematic evaluation of graph-enhanced LLMs (GraphRAG) against base models (Llama)
- *Flowchart Comprehension*: Capability assessment across vision-language models (GPT-4V, Qwen-VL)
- *Climate Entity Recognition*: Domain adaptation analysis comparing general-purpose LLMs to climate-focused variants (ClimateGPT)
- *Entity Linking*: Hybrid approach evaluation contrasting graph-neural methods with traditional string-matching baselines

3. Adaptive IE Architectures Developed task-specific frameworks with measurable improvements: *KG-RAG*: Reduced climate entity hallucinations through dynamic knowledge graph constraints. *FlowLearn*: Poposed using modular pretraining (caption analysis $\rightarrow$ visual decomposition $\rightarrow$ semantic grounding).

These contributions establish a scalable framework for continuous knowledge integration, providing researchers with a practical scholarly infrastructure for both retrospective synthesis and forward-looking discovery.

CHAPTER 2

REVIEW OF LITERATURE

2.1 Scientific Knowledge Infrastructure

2.1.1 Publication Management

In the realm of publication management, climate science lacks comprehensive resources for organizing related publications, especially when compared to other scientific fields. For instance, the medical field benefits from PubMed [1], and the computer science domain has repositories like Papers With Code[2].In contrast, climate science relies on general scientific databases such as Scopus[3] and Web of Science[4], which cover a broad range of disciplines but are not specifically designed for climate research. For climate-specific, the Coupled Model Intercomparison Project (CMIP) Publication Hub[5] lists 394 publications–a small fraction of the climate science literature. Europe PMC[6] hosts a database of over 44.2 million life sciences publications but lacks direct links to full papers. While Europe PMC's SciLit allows users to link in-text mentions to entities in its vocabulary, its reliance on human input leads to sparse annotations. Additionally, annotations are limited to abstracts, and many papers lack annotations altogether.

[1] `https://pubmed.ncbi.nlm.nih.gov`
[2] `https://paperswithcode.com`
[3] `https://www.scopus.com`
[4] `https://www.webofscience.com`
[5] `https://cmip-publications.llnl.gov`
[6] `https://europepmc.org`

2.1.2 Scientific Information Extraction Datasets

The systematic transformation of scientific literature into structured knowledge bases has been enabled by domain-specific corpora spanning biomedicine, computer science, and interdisciplinary research. In biomedicine, MedNER [8] provides granular annotations for disease mentions across clinical narratives, while bioNerDS [9] identifies database and software references in bioinformatics' literature. Computer science resources exhibit complementary specialization: SemEval-2017/2018 [10, 11] establish benchmarks for research concept tagging, and NLP-TDMS [12] introduces temporal annotations for method evolution tracking. These corpora collectively address domain-specific lexical challenges—from biomedical neologisms (*BRCA1 mutation"*) to computational vernacular (*transformer architectures"*)—through annotation schemas encompassing atomic entities, coreferential chains, and cross-document relationships.

2.1.2.1 AI-Centric Textual Corpora

Contemporary AI literature mining relies on three foundational resources, each addressing distinct facets of methodological reporting. SciERC [13] establishes foundational annotations across 500 scientific abstracts, identifying six entity categories (*Task, Method, Metric, Material, Other-ScientificTerm, Generic*) alongside coreference chains and binary relations. SciREX [14] extends this paradigm to document-level IE, introducing multi-task annotations for salient entity detection and cross-sentence relationship extraction. TDMSci [15] narrows focus to NLP

methodology, annotating 2,000 sentences for *Task*, *Dataset*, and *Metric* mentions while omitting method entities—a critical limitation for comprehensive workflow analysis.

2.1.2.2 Climate Science Related Resources

Contemporary climate science corpora systematically exclude three critical modeling constructs: (1) *experimental protocols* (e.g., CMIP6 scenario specifications), (2) *observational variables* (e.g., aerosol optical depth), and (3) *teleconnection patterns* (e.g., El Niño-Southern Oscillation phase transitions). Domain-agnostic benchmarks exacerbate this gap by prioritizing generic entities like *Dataset* and *Location* over climate-specific technical lexicons, limiting computational workflow analysis.

Existing Climate Corpora Existing structured resources for climate knowledge predominantly target policy analysis and impact documentation. The CPo-CD Dataset [16] exemplifies this trend, annotating 13,728 short text segments (2–250 words) with policy elements such as *Target*, *Action*, *Policy*, and *Plan*. Similarly, CLIMATELI [17], the first manually annotated dataset for climate entity linking, maps 3,087 entity spans to Wikipedia across genres like IPCC reports and news articles, though its scope remains constrained to broadly recognized concepts. Efforts to systematize climate impacts [18], who employ LLMs to extract 300 records of extreme events (e.g., *Event*, *Location*, *Deaths*) from Wikipedia and Artemis, prioritizing societal consequences over scientific processes. In the corporate sustainability domain, Usmanova and Usbeck [19] transform 124 reports into a

knowledge graph with ontology classes like *Organization* and *Risks*, alongside descriptive relations such as *hasDescription*, while Garigliotti [20] combines LLMs with retrieval-augmented generation (RAG) to classify sustainability targets in 33 reports. Though these resources advance policy tracking and corporate disclosures, they overlook technical climate science entities fundamental to modeling workflows—experiments, observational variables, and teleconnections. Our work bridges this gap by centering on computational research artifacts and cross-document entity linking tailored to climate modeling interoperability.

Climate Taxonomies Numerous vocabularies related to climate science have been organized to facilitate research, including notable examples such as NASA's Global Change Master Directory (GCMD), Semantic Web for Earth and Environment Technology Ontology (SWEET) [21], CMIP6 Controlled Vocabularies (CMIP6CV) [22], and Obs4MIPs [23]. Each taxonomy focuses on different aspects of climate science. GCMD is among the most comprehensive and popular taxonomies, encompassing a wide array of climate-specific entities, including projects, locations, and climate events—attributes crucial to researchers when analyzing publications. In contrast, CMIP6CV and Obs4MIPs are specifically tailored to climate modeling, featuring variables that include specific names of climate experiments and model variable names.

2.1.2.3 Scientific Figure Datasets

In addition to scientific text, figures such as diagrams and charts play an indispensable role in conveying complex information within the scientific community. This section delves into interdisciplinary research at the nexus of computer vision and natural language processing, with a specific emphasis on the comprehension of visual figures in scientific documentation. Our study narrows its focus to a particularly under-explored subset of these figures: flowcharts. These visual tools are pivotal for understanding processes and methodologies in scientific literature, yet they have not been extensively investigated within the research community. Through this targeted approach, we aim to shed light on the nuances of flowchart comprehension and its implications for enhancing scientific communication.

Significant efforts have been made to develop methodologies and datasets aimed at extracting and understanding scientific figures. Notable methods include PDFigCapX [24], PDFMEF [25], PDFFIGURES [26], and PDFFIGURES2 [27], which facilitate the extraction of figures, captions, and related information from scholarly articles. Datasets like VIS30K [28] and PDFFIGURES2 [27] advance figure extraction by providing detailed annotations concerning figure locations. Moreover, ACL-FIG [29] and DocFigure [30] focus on figure classification, enhancing the understanding of various figure types, including bar charts and architecture diagrams. Additionally, specialized datasets like SciCap [31] and Parsing-AUC [32] concentrate on image captioning and summarization for experimental results figures.

Despite these advances, there is a noticeable gap in datasets that offer detailed annotations specifically for flowcharts. For instance, ACL-FIG [29] includes only about 200 flowcharts categorized under architecture diagrams and neural networks, and CSDia [33] focuses on logical diagrams but lacks detailed caption information as it is not derived from the scientific literature. In contrast, SCI-CQA [34] is a benchmark designed to evaluate multimodal models on scientific charts, including 2,953 flowcharts and 202,760 image-text pairs collected from 15 top-tier computer science conferences over the past decade.

2.1.3 Limitations of Current Resources

Contemporary scientific information extraction resources exhibit four systemic constraints that hinder comprehensive knowledge synthesis.

First, corpus *scale and coverage* remain insufficient for data-intensive modern methods. Climate science exemplars like the CMIP Publication Hub catalog fewer than 400 papers—less than 0.7% of annual climate literature—while AI-centric corpora like SciERC and TDMSci average just 500–2,000 annotated instances. This paucity of training data forces models to extrapolate from narrow samples, as evidenced by high error rates on rare climate variables like PM transport coefficients in pilot studies.

Second, *lexical diversity gaps* persist across annotation schemas. TDMSci's exclusive focus on capitalized dataset names (e.g., *GLUE*, *COCO*) inadequately represents real-world usage patterns, which often feature mixed casing. Similarly,

climate policy corpora like CPo-CD emphasize broad concepts (e.g., *Target*, *Action*) while overlooking technical descriptors such as CMIP6 experiment codes (`ssp245`).

Third, *entity linkability limitations* hinder cross-study knowledge integration. Although CLIMATELI includes 3,087 Wikipedia links, only a small proportion of them pertain to climate science research, limiting the corpus's effectiveness in connecting domain-specific entities and supporting comprehensive knowledge synthesis.

Fourth, *multimodal grounding deficiencies* limit figure and workflow analysis. Flowchart datasets like ACL-FIG contain just 200 samples—insufficient to model the 47 distinct visual idioms (e.g., feedback loops, parallel processing) identified in climate model documentation. Moreover, existing taxonomies (SWEET, GCMD) lack mappings between figure elements (e.g., flowchart decision nodes) and their textual method descriptions.

These limitations collectively restrict scientific information extraction (IE) systems to narrow, domain-static applications, highlighting the need for resources that incorporate large-scale annotations, lexical variants, persistent identifiers, and multimodal grounding for broader and more adaptable use.

2.2 Textual Information Extraction Models

2.2.1 Non-LLM Models

Scientific mention detection has progressed through three architectural paradigms. Early work [35] introduced CRF-LSTM hybrids for sequence labeling, utilizing

contextual windows to identify dataset mentions. Building on this, subsequent research incorporated pre-trained embeddings, with BiLSTMs combined with ELMo vectors achieving improved performance on computer science abstracts [36]. Further advancements demonstrated BERT's effectiveness in capturing nested method mentions [37]. More recently, the NLP-TDMS system [12] expanded this line of work by employing transformer-based models for enhanced mention detection.

Multi-task learning frameworks address annotation sparsity by jointly optimizing related objectives, such as entity linking and relation extraction. For instance, SciREX [14] proposed a multi-task model based on BERT that performs scientific mention identification, salient mention detection, pairwise coreference, and salient entity clustering, achieving state-of-the-art performance on the SciREX dataset.

2.2.2 LLM-Based Models

Large language models (LLMs) exhibit exceptional performance in scientific information extraction (IE) tasks, particularly those requiring schema induction from implicit context. In chemical entity recognition, LLaMA-3-70B achieves 93% accuracy in extracting synthesis protocols for reticular materials [38]. Similarly, for biomedical knowledge extraction, GPT-4 attains 87% accuracy on literature concerning HIV drug resistance [39]. Other specialized models have further advanced scientific IE. REBEL [40] utilizes BART-large, fine-tuned for relation extraction, effectively identifying relationships between entities. GPT-NER [41] leverages GPT-3 for named entity recognition (NER) in a zero-shot setting, allowing it to perform extraction without domain-specific training data. Likewise, PromptNER [42] employs GPT-

4 with prompt engineering for effective NER, making it adaptable across various contexts.

Beyond domain-specific approaches, emerging solutions enhance LLM outputs through structured knowledge integration. GraphRAG improves factual consistency by grounding LLM responses in dynamic knowledge graphs, leveraging schema-constrained decoding to reduce hallucination rates. LightRAG [43] incorporates graph structures into text indexing and retrieval, reducing computational costs and enabling rapid domain adaptation. Its incremental update algorithm ensures timely integration of new data, maintaining effectiveness in evolving environments. SAC-KG [44] automates large-scale knowledge graph construction, scaling to over one million nodes with an 89.32% triplet accuracy. Meanwhile, VEGGIE [45] employs context-sensitive graph grammars for error-correcting parsing, enhancing the robustness of extracted information. CollabKG [46] serves as a toolkit for cooperative human-machine information extraction, facilitating knowledge graph construction. Additionally, TechGPT-2.0 [47] is designed for technology-oriented tasks, including automated knowledge graph building.

These advancements underscore the growing synergy between LLMs and knowledge graph-based frameworks, offering promising avenues for more accurate and adaptable scientific IE systems.

2.2.3 Model Limitations in Domain-Specific Information Extraction

Three systemic limitations constrain model effectiveness in domain specific IE.

First, hallucination risks intensify in domains requiring strict terminological precision. For example, climate science IE models may conflate distinct concepts such as "RCP8.5" and "SSP5-8.5," while biomedical models may confuse drug nomenclature variants (e.g., "EGFR-TKI" vs. "ALK-TKI" kinase inhibitors). Techniques like contrastive decoding [48] mitigate this by suppressing implausible token sequences, but they struggle with climate science's long-tail concepts absent from general pretraining corpora.

Second, domain mismatch persists even in adapted models like SciLitLLM [49], which focuses on broad scientific literature rather than climate-specific discourse. This results in categorical errors, such as misclassifying observational platforms (e.g., "Argo floats" as geographic locations) or mislinking abbreviations (e.g., "ENSO" to entertainment entities).

Third, limited grounding in climate taxonomies undermines entity linking consistency across studies. While RAG partially addresses this [20], current implementations prioritize policy targets over technical modeling artifacts. ClimateIE addresses these gaps via structured annotations and hybrid human-LLM curation pipeline, enabling robust grounding of climate entities while minimizing hallucination risks.

2.3 Visual Understanding Models

2.3.1 Image Captioning

Research in image captioning [50, 51], particularly in scientific chart image captioning [52], has seen significant advancements, exemplified by works such as Parsing-AUC [32], which combines figure semantics extracted via OpenCV with textual information from the main text to generate comprehensive figure summaries for AUC figures.

2.3.2 Visual Question Answering (VQA)

The field of VQA [53, 54, 55, 56] has seen substantial advancements, with datasets like VL-ICL Bench [57] providing benchmarks for multimodal in-context learning. In scientific contexts, CSDia [33] employs models based on Diagram Parsing Nets to address VQA tasks, particularly in diagram analysis. Specifically designed for scientific result figures, FigureSeer [58] excels in figure localization, classification, and analysis, enabling detailed indexing and result summarization. Notably, FigureSeer incorporates an intermediate figure parsing step to extract key components such as axes, legends, and data points. This highlights the importance of figure decomposition in improving scientific figure comprehension, particularly for flowcharts.

In the medical domain, specialized VQA models have emerged to address challenges in interpreting radiology and clinical images. CGMVQA [59] focuses on medical VQA for radiology image analysis, while MedFuseNet [60] enhances multimodal learning through an attention-based framework. Similarly, MMBERT [61]

leverages pre-trained BERT models for language processing, adapted specifically for medical VQA, and has been referenced in several survey studies.

Beyond scientific and medical applications, VQA research has explored innovative approaches to improving answer generation and interpretability. Neural-Symbolic VQA [62] and αILP [63] integrate neural network outputs with symbolic reasoning, facilitating structured answer formulation. Additionally, work on structure-aware visualization retrieval [64] underscores the significance of incorporating structural information into VQA systems. Their findings suggest that users prefer similarity evaluations based on visual structure rather than low-level pixel comparisons, advocating for a deeper, more semantic approach to image understanding.

These developments collectively showcase the evolution of VQA, from domain-specific models to versatile, multimodal architectures capable of handling complex visual and textual interactions.

2.3.3 Large Vision-Language Models (LVLMs)

The development of Large Vision-Language Models (LVLMs) has significantly advanced visual understanding by integrating sophisticated vision techniques with large language models (LLMs). These models learn from both images and text simultaneously, enabling them to tackle various multimodal tasks such as visual question answering (VQA) and image captioning. The OpenCompass Multi-modal Leaderboard[7] ranks leading LVLMs, including notable models such as GPT-4V [65],

[7]https://rank.opencompass.org.cn/leaderboard-multimodal

Gemini [66], LLaVA [67], Claude [68], InternLM [69], Qwen-VL [70], Step-1V[8], and DeepSeek [71].

Transformer-based multimodal architectures have further expanded the capabilities of LVLMs in VQA and related tasks. ViLT [72] employs a transformer-based vision-language framework, demonstrating strong performance in VQA. Similarly, LLaVA [67] extends these capabilities as an open-source VQA model adaptable across diverse image types. PaliGemma [73], developed by Google, introduces a versatile multimodal vision-language model, further enhancing VQA applications. Meanwhile, GPT-4o [74] from OpenAI processes both images and text, enabling advanced vision-language interactions across multiple domains.

These models are trained on diverse multimodal datasets, including those for VQA, optical character recognition (OCR), and academic-related VQA, all of which are crucial for scientific flowchart understanding.

2.3.4 Image Decomposition

Several works have focused on extracting objects and their relationships from scientific figures. Notably, [75] pioneers the conversion of scientific equation images into LaTeX format. ChartDetective [76] introduces an interactive application for converting result chart images into SVG, preserving semantics and component relationships. However, it is essential to note that this approach relies on user interaction and takes about 4 minutes for a single conversion. For non-scientific

[8]https://www.stepfun.com/

domains, Flow2Code [77] converts hand-drawn flowcharts to simple code by using object detection and rules.

CHAPTER 3

DMDD: A LARGE-SCALE DATASET FOR DATASET MENTIONS DETECTION

Having established the limitations of current resources through our literature review, we begin our investigation of scientific information extraction with the foundational task of dataset mention detection. The reliable identification of these mentions underpins critical downstream tasks including attribution analysis, reproducibility validation, and knowledge graph construction. However, as discussed before, existing resources remain constrained by three systemic limitations: narrow scope (e.g., TDMSci's exclusive focus on NLP methods [15]), incomplete annotation schemas (e.g., bioNerDS' exclusion of versioning information [9]), and inadequate scale (median 1.2k instances across surveyed corpora).

We present **DMDD** (Dataset Mentions Detection Dataset), addressing three core limitations of prior work:

- **Granularity**: Precise annotation of dataset names (e.g., *ImageNet*) while excluding pronominal references (e.g., "the dataset") or underspecified material mentions (cf. SciREX's material entities [14])
- **Scale**: significantly larger than prior domain-agnostic benchmarks, with 380K+ mentions across 26K scientific articles

- **Contextual Richness**: Full-text annotations capturing mention contexts critical for disambiguation (e.g., *"We evaluated on ImageNet-1K (Russakovsky et al., 2015)"*)

In the following sections, we detail DMDD's corpus development, including our hybrid human-AI annotation protocol and quality control measures. Figure 3.1 illustrates a representative data entry with parsed article structure and mention spans. We also introduces DMDD-Eval, a manually-curated benchmark for cross-domain generalization testing, while providing a systematic comparison against existing resources across different dimensions.

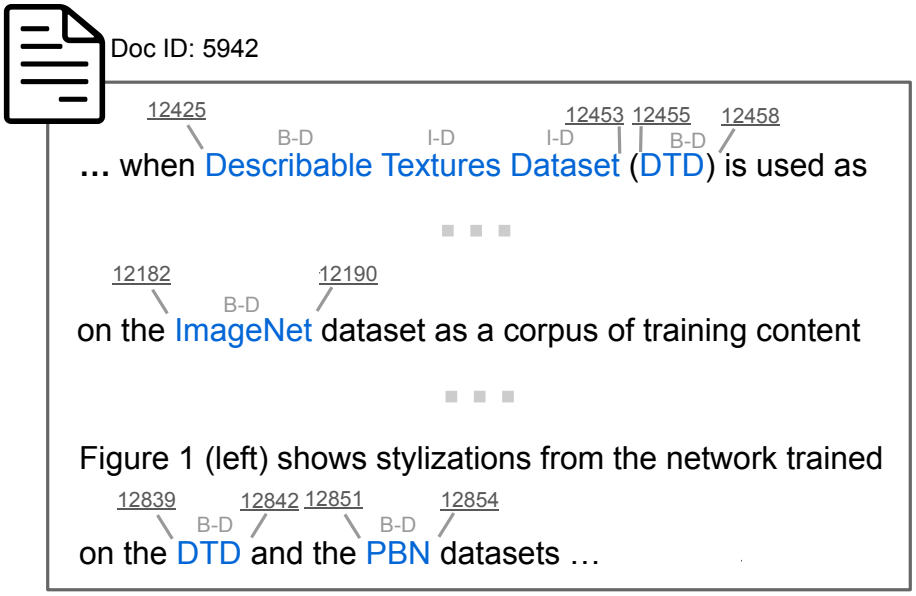


Figure 3.1: Example of paper-level annotation in DMDD. We mark each occurrence of dataset (D) in papers and give the in-text spans. We can generate the BIO annotation. For example, the dataset mention 'ImageNet' spans 12182 to 12190 and has a BIO tag as 'B-D'.

3.0.1 Data Collection

We built DMDD's main corpus by combining data from S2ORC [78] and Papers with Code (PwC). The parsed scientific articles are obtained from S2ORC [78], which is a dataset based on the Semantic Scholar website. S2ORC is a unified resource that combines aspects of citation graphs (i.e., rich paper metadata, abstracts) with a full-text corpus that preserves important scientific paper structure (i.e., sections, inline citation, references to tables and figures). For in-text level annotation of dataset mentions, we used distant supervision to derive the annotations from existing data sources with document-level annotation. We sourced the document-level annotation from Papers with Code (PwC), which is a free and open-source website with machine-learning papers, code, datasets, methods, and evaluation tables. For each available paper listed in PwC's data files, we obtained the publication details, PDF web links, and links to related GitHub code. Most of these publication details are edited by the authors of those papers. However, the information about datasets mentioned in the papers is not organized for download. To obtain such information, we conduct web scraping of the 'Dataset Section' of each paper's webpage in PwC, which contains human annotations on the document-level about the datasets mentioned in the paper.

3.0.2 Annotation Procedure

We describe our distant supervision procedure to create the in-text mention annotation for dataset mentions in this subsection. The document-level annotations are based on the data provided by PwC's users. Our premise is that we can take

the names supplied by authors in PwC and match them in the main text of a paper. For the most part, this is a correct assumption. However, users do not often give complete information about the artifacts used in their papers. For example, they may only give a partial spelling of an entity name (e.g., 'CIFAR' instead of 'CIFAR10') or use a different spelling (e.g., 'CIFAR-10' in PwC and 'CIFAR10' in the paper). Thus, we cannot proceed with a strict matching procedure of dataset names collected from PwC in the text of the papers.

We commence by creating a dictionary that defines all dataset entities in DMDD. For each dataset entity, we store the following information: name, full name, and web page link in PwC. Next, we create regular expressions (regex) for each dataset entity. The regular expression creation process is described in detail in Section 3.0.3. We use regex as an approximate matching procedure to label the parsed text of a paper. Data engineers refer to such data labeling rules as *labeling functions* [79]. Two example DMDD dictionary entries containing its regex can be found in Figure 3.1.

Using the document-level annotation on dataset mentions and the regex, we annotated 31,219 scientific articles. For each article, we have the concatenated full-text, section span, document-level dataset annotations, in-text dataset mention span, and the entity index for each mention. Example data can be visualized in Figure 3.1. In addition, we also store section information for each document, which includes the section names and their corresponding starting and ending indices in the concatenated full text. The reason we include section span is that we believe 'section' may provide additional semantic information and can impact the detection accuracy. For example,

a detection algorithm should be more sensitive to candidates in experiment sections where authors typically describe their datasets.

3.0.3 Regular Expression Rules

The regex objective is to incorporate as much variety in dataset mentions as possible. However, we do not seek to have an optimal regular expression. First, such a rule is difficult to create manually, and second, we seek to generate enough (weak labeled) data to enable training NER recognizer. Instead of constructing regex for each dataset individually, we use a set of rules to construct regex for all dataset entities, using their short name and full name listed in PwC as base names.

For the 6,675 dataset entities listed in the PwC dataset definition file, there are 8,708 listed name variants. Using an exact match with the base names, we match 7,989 variants. These matched variants are just the short names and full names of the entities. To enhance the exact match, we used a set of rules to customize the regular expression for each base name. The number of additional variants matched with the added rule compared to the exact match is shown as #Matched.

1) We allow optional space and '-' between words. For example, dataset entity 'CIFAR-10' may be mentioned as 'CIFAR 10' and 'CIFAR10' in papers. To allow such variation, we customize the regex as 'CIFAR-*\s*10'. (#Matched = 77).

2) We create acronyms for names including multiple words by combining the initials of the words. For example, we create an acronym 'WTQ' for entity 'WikiTableQuestions'. (#Matched = 14).

3) We ignore casing for units appearing in names. In particular, if '3D'/'3k'/'3m' in names, we allow matching '3d'/'3K'/'3M'. For example, dataset entity 'DBP15K' may be mentioned as 'DBP15k' in papers. To allow such variation, we customize the regex as 'DBP15[Kk]'. (#Matched = 4).

4) We allow optional decimal places for versions and numbers. For example, the dataset entity 'OntoNotes 4.0' may be mentioned as 'OntoNotes 4'. To allow such variation, we customize the regex as 'OntoNotes 4\.*[0-9]*'. (#Matched = 5).

5) We ignore case for words that have a length greater than 4 and the lowercase of the name is not a common English word. For example, we ignore cases when matching for dataset entity 'SciREX', so that it matches 'SCIREX' and 'scirex' that may appear in text. We enforce case matching for dataset entity 'SHAPES'. (#Matched = 286).

6) We allow optional suffixes including 'ing' and 'ion'. For example, the dataset entity 'Deep Soccer Captioning' may be mentioned as 'Deep Soccer Caption'. To allow such variation, we customize the regex as 'Deep Soccer Captioni*n*g*'. (#Matched = 0).

7) We allow optional plural forms including 'es' and 's'. For example, the dataset entity 'MovieLens' may be mentioned as 'MovieLen' in papers. To allow such variation, we customize the regex as 'MovieLens*'. (#Matched = 0).

While PwC's listed variants do not include the patterns from rules 6 and 7, we observe many such variations in DMDD's papers caused by typos and loose writing.

Using all the rules outlined above, we identify names with the corresponding patterns and customized the regex accordingly. This final set of customized regex allows us to cover most of the listed variants, leaving us with only 74 unmatched

variants. To address these unmatched variants, we use them as the base names and created additional regex for their corresponding entities.

3.0.4 Data Preprocessing

With the help of the SpaCy python library, we convert the original annotation in the span format to BIO format. After the first stage of preprocessing, we discover that we miss some of the annotations for dataset mentions in some sequences. This is because, on PwC websites, the authors or the editors often only annotate the datasets being used in experiments while missing the ones being mentioned. The missing mentions can introduce bias in training as the models may be negatively impacted by learning about the false negative. Thus, in the second stage of preprocessing, in order to reduce the number of missing annotations, we combine all regex to search for all possible mentions of the dataset entities in DMDD's dictionary, which was obtained from the PwC website. We exclusively apply the second stage of preprocessing to sentences that contain detected dataset mentions from the first stage. This limits the addition of mentions to contexts where the occurrence of dataset mentions is highly likely; this helps mitigate false positives arising from ambiguous entities, such as 'SGD'. While 'SGD' often appears as a method name in sentences without dataset mentions, it can also appear as a dataset name in co-occurrence with other dataset mentions.

To ensure a consistent comparison between our proposed corpus and existing corpora, we adopted a consistent data preprocessing strategy across all related corpora used in our experiments. In the case of NLP-TDMS and RCC, we used the original

Corpus	Inst. Unit	# Inst.	# Mentions	# Unique Mentions	# Unique Entities	Entity Linking
DMDD (ours)	paper	31,219	449,798	10,807	6,675	explicit
SciERC [13]	abstract	164 (69)	770 (122)	644 (116)	-	-
SciREX [14]	paper	407	10,548	2,857	-	-
NLP-TDMS [12]	paper	153	1,164	67	99	explicit
TDMSci [15]	sentence	445	612	478	-	-
bioNerDS [9]	paper	60	920	145	-	-
RCC	paper	2,256	36,597	1,345	1,028	explicit
Heddes [80]	sentence	2,664	3,416	2,319	-	-

Table 3.1: Summary of corpora for dataset mention detection. The numbers in the brackets for SciERC relate to the corrected version of SciERC without annotation errors.

text of each paper and their corresponding dataset mention list to develop similar regex patterns to extract dataset mentions in BIO-format. For bioNerDS, the dataset mention span annotations were already provided in BIO-format, so no additional processing was necessary. For SciERC, SciREX, and Heddes, the sequences were already provided in BIO-format annotation.

3.1 Evaluation Set with Human Annotations

We manually annotated two sets of instances for evaluation purposes, one set is from DMDD and the other is from SciREX. As SciREX provides publicly available manually annotated documents with scientific entities, we only needed to refine their annotations to meet DMDD's standards. All evaluation sets were manually annotated by three NLP researchers using brat rapid annotation tool [81]. We aggregated the annotations by keeping the mentions where at least two annotators agreed.

For the DMDD evaluation set (DMDD-E), annotators were tasked with manually annotating 450 papers that were sampled from DMDD's test set. The annotators were instructed to verify the detected mentions from DMDD's main corpus and identify any

missing mentions in each paper. Additionally, they were required to verify the entity linked to each mention. To ensure accuracy, annotators were directed to search the PwC website and Google to confirm dataset entities during the annotation process.

To assess the level of agreement between annotators, we used the relaxed span matches method, which considers a match when the dataset mention spans from the three annotators overlap. The resulting Fleiss kappa of 0.79 represents a substantial agreement between annotators. DMDD's evaluation set contains 13,039 mentions for 682 DMDD entities, with 1,964 mentions that could not be linked to the DMDD dictionary. On average, each annotator required approximately 15 minutes to annotate a pre-annotated paper with weak labels.

When compared to DMDD's evaluation set, the weak labels from DMDD's main set obtains an F1 score of 77.9%, recall of 68.1%, and precision of 91.2%. The low recall indicates that most of the weak labeling errors are due to missing dataset mentions. We identify two main reasons for the missing mentions. First, mentions may contain rarely-used version names that distant supervision provides only partial annotation for, such as 'KITTI 2012', where only 'KITTI' is tagged and the version part of the name, i.e., '2012', is ignored. Second, missing mentions may occur in contexts without mentions of the document-level annotated dataset, such as in related work sections where only one dataset is mentioned, or in sentences where the dataset is mentioned by itself as a pre-trained dataset in the description of methods.

3.2 Comparison with Related Corpora

We compare DMDD with seven related corpora containing dataset mentions annotations in Table4.1, where 'Inst.' is used to represent 'instance' and '#' is used to represent 'number'. In order to compare corpora fairly, we exclude the negative instances from the calculation of '# Instances', as some corpora do not contain negative instances.

3.2.1 Corpora Size

DMDD has the largest size among the discussed corpora, in terms of the number of instances (# Inst. = 31K), instance unit (Inst. Unit = Paper), and the number of mentions (# Mentions = 450K). With paper-level annotations, DMDD allows for a larger input unit, such as a section, which can provide richer context and potentially benefit mention detection models.

SciERC samples instances from abstracts. Sampling instances from a specific section of papers may create corpora with limited variation in lexical and syntactic expressions (for example, the language of abstract sections is different from that of methodology sections). A benefit of DMDD over most of the other existing corpora is that an entity mention appears in multiple sentences across the 31K papers, offering diverse context learning opportunities in training. This is captured by the number of unique mentions and the number of mentions in Table 4.1. While the related corpora give better-labeled data (because they're manually created), their data annotation processes are not scalable since they heavily depend on manual labeling.

Corpus	Long Mention		Alpha. & Punct.		All Lower	
	#	%	#	%	#	%
DMDD	3,044	28	7,903	73	1072	10
SciERC	552	86	612	95	353	55
SciERC$_C$	7	10	50	72	1	1
SciREX	2,122	74	2,102	73	307	11
NLP-TDMS	48	48	60	61	0	0
TDMSci	335	70	317	66	10	2
bioNerDS	34	31	104	95	3	3
RCC	2,869	91	2,469	78	71	2
Heddes	2,161	83	1,774	68	81	3

Table 3.2: Distribution of different types of dataset mentions in DMDD and existing corpora. # and % indicate the number and percentage of the corpus' unique mentions exhibiting certain characteristics. SciERC$_C$ represents the corrected version of SciERC without annotation errors.

3.2.2 Diversity of Dataset Mentions

Intuitively, dataset names (e.g., 'CIFAR10') that consist of a single word, that include capitalized letters, and do not have non-literals are easy to detect. However, many dataset names do not follow this pattern. They may contain non-literals (e.g., 'YUP++'), may not be capitalized (e.g., 'iris'), or may contain multiple words (e.g., 'Atomic Visual Actions'). Such diversity of dataset naming poses detection difficulties. A (training) corpus needs to avoid being biased toward any of such categories and contain enough samples from each category. We perform an in-depth analysis of all annotated dataset mentions in related corpora to examine the diversity of mentions.

For each corpus, we perform the following evaluation steps and summarize the evaluation results in Table 3.2. First, we extract all in-text mentions of the dataset names, using the provided annotations. We derive the unique mentions from all the in-text mentions. Notably, unique mentions do not equal unique datasets as one

dataset may be referred to as different text strings (e.g., 'MHP' may be referred to as 'Multiple-Human Parsing'). Second, we find mentions with different characteristics, which are defined as follows.

1) Long mentions. If the mention contains white spaces, then it is a long mention containing multiple words. This is important as long mentions are often harder to be detected accurately than single-word mentions.

2) Character level composition. Alphabet and Punctuation Only (Alpha. & Punct.): check if the mention contains only alphabet and punctuation. We want to see the number of mentions containing no numerical characters. From a reader standpoint, it is often easier to classify entities with a combination of alphabets and numerical values (e.g.: 'MediaEval2010') as dataset names than those without (e.g.: 'English-Hungarian').

3) Capitalization. All Lower-cased (All Lower): We seek to account for dataset names with all the characters being lowered-cased in a dataset mention. As commonly agreed, words including upper case letters often indicate that they are specialized words and are more likely to be dataset mentions than those without upper case letters.

As shown in Table 3.2, with the exception of DMDD, NLP-TDMS, and bioNerDS, the available corpora demonstrate an imbalanced distribution that skews towards long mentions. SciERC and bioNerDS, in particular, exhibit a prevalence of mentions that consist solely of letters and punctuation, with only a small fraction containing numeric characters. Additionally, with the exception of SciERC, all corpora are inclined towards mentions that feature uppercase letters. Hence, individually none of them

have enough unique mentions from each category to enable training of robust models across all categories. We also note that the characteristics presented in Table 3.2 are non-exhaustive, non-exclusive, and may overlap.

3.2.3 Entity Linking

Entity Linking (EL) for datasets is the task of associating a dataset mention in text with a dataset entity in a knowledge base, such as Papers with Code. The entity linking information for dataset mention is important as it enables users to refer to the right dataset or download the correct dataset for empirical studies. We distinguish two categories of linking: explicit linking and non-linking. We categorize the type of linking for existing corpora in Table 4.1. We note that in Table 4.1, the "-" symbol represents non-linking.

DMDD is created based on PwC and each entity mentioned in DMDD's main corpus has an explicit link to the PwC website with a unique identifier. RCC and NLP-TDMS also have explicit linking since they provide URL links to the knowledge bases with dataset information. Specifically, all the datasets from RCC can be linked to ICPSR[1] and all the datasets in NLP-TDMS can be linked to NLP-Progress[2]. However, all the other corpora do not provide such explicit linking information.

For the related corpora without explicit linking information, we attempted to link their annotated mentions to PwC and the other websites, like the ACL Anthology, but we were unsuccessful in linking a significant portion of the annotated mention. In addition, our early empirical studies with these corpora showed an unexpectedly low

[1]https://www.icpsr.umich.edu/web/pages/
[2]https://nlpprogress.com/

Corpus	Examples
DMDD	'MNIST', 'General Language Understanding Evaluation benchmark'
SciERC	'image data', 'written texts',
SciREX	'SQuAD)', 'augmented PASCAL train set'
NLP-TDMS	'SemEval-2010 Task 8', 'Quora Question Pairs'
TDMSci	'forums', 'a separate set of 40 ACE 2005 newswire texts'
bioNerDS	'String', 'Gene Ontology'
RCC	'balance sheet data', 'External Position Report'
Heddes	'MNIST or the ImageNet dataset', 'text datasets'

Table 3.3: Dataset mention annotation examples from DMDD and existing corpora.

recall rate on detecting dataset mentions, which prompted us to manually verify some of the data. We asked two Ph.D. students with NLP expertise to manually go over the annotated data in SciERC. It was not our goal to verify all data sources, which would have taken substantial labor. Table 3.3 shows some example dataset mention annotations for related corpora. We identify four potential reasons contributing to the failure of linking. We exemplify them using mentions from SciERC.

1) Mentions include extra characters or text strings [9 (1%)]. For example, the mention 'aligned wordnets' includes the descriptive text 'aligned' for the datasets. Additionally, in the original document, this mention actually refers to multiple wordnets that are being aligned by the proposed method. In Table 3.3, 'SQuAD)' includes the extra character ')' which may be the result of human error.

2) Mentions include more than one dataset [8 (1%)]. For example, 'SemCor and Senseval-3 datasets'.

3) Mentions do not include the actual dataset name [559 (87%)], for example, 'records' and 'CD-covers'. This is because some related corpora are annotated with pronominal reference to entities, as defined in ACE 2005 [82]. Pronominal reference is not helpful in linking mentions to dataset entities, especially when the corpora are not annotated on the paper level and the proper name reference is missing from the annotated instance. Within this characteristic group, there are also confusing mentions not using the most commonly-used dataset names or missing part of the names [5 (1%)]. For example, 'treebank' can denote many possible datasets, such as The Penn Treebank [83] and CHILDES Treebank [84]. This further points toward the need to include linking attributes in the annotation whenever possible.

Among all of the unique mentions in SciERC, only 69 (11%) do not exhibit the three discussed characteristics. As shown in Table 4.1, when only considering the correct mentions, the number of mentions and instances with mentions are significantly reduced. Also, as shown in Table 3.2 for SciERC$_C$, the percentage of long mentions and all-lower-case mentions drops significantly, yielding a more biased set of dataset mentions.

All existing corpora, except NLP-TDMS, share similar characteristics to SciERC. NLP-TDMS follows the NLP-Progress taxonomy website to annotate their entities, which means all the dataset names they used for labeling their instances are actual dataset names.

In contrast to the existing corpora, DMDD has the following advantages. DMDD is the largest corpora with more than 31K instances. DMDD has the largest number of mentions and the largest number of unique mentions, providing more mention examples than existing corpora. In terms of the diversity of dataset mentions, DMDD exhibits some biases on having a small percentage of all-lower cased mentions. However, since DMDD contains a significantly larger amount of mentions and unique mentions than existing corpora, DMDD can still provide enough examples with different characteristics. In terms of entity linking, all DMDD's annotated mentions can be directly linked to Papers with Code web pages.

3.3 Experimental Setup

The experiments are designed to address the task of dataset entity mentions and entity linking, with three primary objectives in mind: establishing baseline performance on our dataset, providing insights into the difficulty of each task, and evaluating the effectiveness of using DMDD for training.

3.4 Mention Detection

We formulate the task of dataset mention detection as a token-level tagging task, and evaluate a broad range of models as baselines in our experiments. To explore the impact of input size, we evaluate models with different input lengths. Since most existing approaches for dataset mention detection operate at the sentence-level, we split the models into two categories: sentence-level models and beyond sentence-level models.

3.4.1 Sentence-Level

We conducted experiments on sentence-level inputs using various models, including Conditional Random Fields (CRF), Bidirectional Long Short-Term Memory (BiLSTM), BERT [85], and SciBERT [86]. For the CRF model, we used features that incorporate Part-of-Speech (POS) tags and keywords [80].

For BERT and SciBERT, we used the pretrained weights: base-cased BERT [85] and scivocab-cased SciBERT [86]. Then, we fine-tuned them on our training corpora. All hyperparameters used for training the models were the same as in the original SciBERT [86], except for the batch size, which was set to 16.

For BiLSTM, we evaluated two additional variations: BiLSTM-G and BiLSTM-W, which utilize pre-trained embeddings initialized with GLoVe [87] and Word2Vec [88], respectively. We loaded both pre-trained embeddings using the Gensim Python library and initialized tokens that were not mapped with pre-trained embeddings to zeros. The embedding layer was updated during training for all tokens. To ensure a fair comparison, we used a 300-dimensional embedding layer for BiLSTM, BiLSTM-G, and BiLSTM-W.

For BiLSTM-G, we used the embedding trained on Wikipedia and Gigaword, converting 30,428 tokens in the entire corpus, while 120,190 tokens were missing from the pre-trained embeddings. We observed that most dataset names were missing from the pre-trained embeddings.

Similarly, for BiLSTM-W, we used the embedding trained on Google News, converting 63,321 tokens while 87,297 were missing. We hypothesize that by

incorporating additional learned semantic information from large corpora, these two versions of BiLSTM can outperform the regular BiLSTM in predicting dataset name mentions.

3.4.2 Beyond Sentence-Level

To evaluate model input sizes beyond sentence-level, we examined two models optimized for longer sequence length: SciBERT and LongFormer [89]. Additionally, we evaluated two different input sizes, section-level and 512-tokens-level. For the section-level inputs, we cropped the documents based on their sections, whereas for 512-tokens-level inputs, we cropped the documents to sequences with a fixed length of 512 tokens. Notably, some of these sequences contain dataset mentions while others do not.

3.5 Entity Linking

Entity linking (EL) for dataset entities, as a special subproblem of EL, differs from the typical general EL task, which links general entities into a huge knowledge base (KB) like Wikipedia. In our work, we utilize PwC as the KB, which contains 7,795 entities. To evaluate the EL task on our dataset, we conduct baseline experiments for EL using two methods. Specifically, we consider the EL given true spans, then we take the span of the dataset mention as the query, and PwC as the KB. We then utilize an information retrieval approach to retrieve the top K most relevant dataset entities in the KB. We conduct experiments in both sparse retrieval and dense retrieval using Pyserini [90]. In Pyserini, sparse retrieval is based on BM25 and uses bag of word

representations, while dense retrieval employs transformer-encoded representations, with the encoder being ColBERTv2 [91]. All parameters use the default settings of Pyserini.

3.6 Train-Test Split

For DMDD and all the corpora used in our experiments, we first perform a train-test split at the document level. Subsequently, we perform a train-test split at other levels, such as section-level and sentence-level, based on the document-level split. For DMDD, we used 70% of the documents for training and 30% for testing.

The DMDD-E set, which is a manually annotated test set of 450 documents, was sampled from the DMDD's test set. We report results on this set in our paper. The DMDD-E set contains a zero-shot subset consisting of 10 dataset entities. These zero-shot entities were randomly selected from DMDD, and none of them appear in any corpus's training set.

When training mention detection models, we use a split of 80% positive sequences and 20% negative sequences in most of the experiments, unless otherwise specified. The goal of negative sentences is to balance the fact that we only consider one type of NER and to facilitate better generalization for deep learning models. In particular, we seek to avoid false positive predictions, since the majority of sentences in scientific papers contain no dataset mentions. Table 3.4 summarizes the median sequence length in tokens and the number of sequences containing dataset mentions in DMDD.

	Median Length	**N. All**	**N. Train**	**N. Test**
Sentence	30	792,554	532,349	260,205
Section	372	245,506	167,954	77,552
512-Token	512	150,207	101,969	48,238
Document	5,729	31,210	21,847	9,363

Table 3.4: The median sequence length in tokens and the number of sequences containing dataset mentions in DMDD.

	Positive and Negative			Positive			Zero-Shot		
# Sentences	10,722			8,602			89		
Model	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
CRF	.681 ± .000	.550 ± .000	.893 ± .000	.682 ± .000	.550 ± .000	.898 ± .000	.342 ± .000	.215 ± .000	.842 ± .000
BiLSTM	.647 ± .013	.546 ± .020	.795 ± .009	.650 ± .014	.546 ± .020	.802 ± .006	.256 ± .020	.168 ± .012	.550 ± .096
BiLSTM-G	.652 ± .012	.548 ± .017	.804 ± .004	.653 ± .012	.548 ± .017	.810 ± .004	.268 ± .041	.181 ± .036	.522 ± .061
BiLSTM-W	.594 ± .019	.498 ± .019	.739 ± .017	.596 ± .017	.498 ± .019	.746 ± .003	.258 ± .037	.175 ± .024	.511 ± .041
BERT	**.751 ± .006**	.635 ± .009	**.920 ± .002**	.753 ± .006	.635 ± .009	**.926 ± .002**	.572 ± .012	.417 ± .012	**.907 ± .004**
SciBERT	**.751 ± .002**	**.639 ± .002**	.912 ± .002	**.754 ± .002**	**.639 ± .002**	.919 ± .002	**.586 ± .008**	**.436 ± .011**	.898 ± .010

Table 3.5: The performance of mention detection models with sentence-level input.

3.7 Experimental Results

This section describes the experimental setup and results of the conducted experiments.

3.7.1 Mention Detection

All mention detection models discussed in Section 3.4 have been trained in 3 rounds with randomly shuffled training sets of DMDD. The average and standard deviation of scores are calculated based on the exact match.

3.7.1.1 Sentence-Level Performance

We evaluate the performance of the mention detection models with sentence-level inputs on three sets: the full set of DMDD-E, the positive subset of DMDD-E, and the zero-shot subset. DMDD-E's full set comprises 80% positive and 20% negative

sequences, with positive sequences including all occurrences in the documents and negative sequences randomly drawn from the documents. Including the negative sequences allowed us to assess the models' ability to accurately classify both positive and negative sentences, which is crucial for real-world applications where the presence of a dataset mention may be rare. Model performance scores, including F_1 score, precision, and recall, were computed based on exact match and are shown in Table 3.5. It is important to note that the relative importance of precision and recall may vary depending on the specific use case and application. For example, precision may be more important in scenarios where false positives can have significant consequences, as it may reduce the reliability of the tool and potentially lead to erroneous analysis or decision-making. On the other hand, in scenarios where missing a dataset mention may lead to missed opportunities for data analysis, recall may be more important.

Overall, SciBERT and BERT performances are close. They have the top performance across all the evaluation metrics in all evaluation sets.

One interesting finding is that the CRF model outperforms the BiLSTM models in our experiments. This can be attributed to the CRF implementation [80], which incorporates expert-designed features that leverage part-of-speech tags and capitalization patterns; they are particularly informative in detecting dataset mentions in scientific literature. In contrast, BiLSTM models rely entirely on learned features, which may not be as effective in capturing the unique nuances of dataset entities. For BiLSTM, the model variation using Word2Vec embedding (BiLSTM-W) and the model variation using GloVe embedding (GloVe) perform similarly to the original version of BiLSTM and bring no significant performance improvement.

Model	F1	Precision	Recall
Sentence-Level Input			
SciBERT	.016 ± .003	.302 ± .059	.008 ± .002
Section-Level Input			
Longformer	.731 ± .004	.625 ± .005	.881 ± .002
SciBERT	**.732 ± .003**	.619 ± .003	**.897 ± .000**
512-Token-Level Input			
Longformer	.695 ± .004	**.661 ± .006**	.733 ± .005
SciBERT	.698 ± .002	.652 ± .006	.750 ± .009

Table 3.6: The performance of mention detection models with different input sizes when evaluating on full documents.

3.7.1.2 Beyond Sentence-Level Performance

For models beyond sentence-level, we crop each evaluated document into overlapped sequences. Specifically, we used a 5% overlap between adjacent sequences. Then, we mapped the predicted results for each sequence back to document-level for evaluation purposes. We used argmax when computing the predicted results for the overlapping tokens. Table 3.6 presents the performance of mention detection models with sentence-level, section-level input, and 512-token-level input on DMDD-E. The table showcases the F1 score, precision, and recall metrics, which are computed based on exact match.

The evaluation of the sentence-level model on entire documents in Table 3.6 shows significantly lower performance than the evaluation on mostly positive sentences containing mentions in Table 3.5. This highlights the challenges of sentence-level models in dealing with the highly sparse dataset mentions in scientific literature.

When considering input sizes beyond the sentence-level, we observed that SciBERT performed comparably to LongFormer. Furthermore, models trained with section-level input have superior performance compared to those trained with 512-

Category	N	F1	P	R
Long Sequences	1,808	0.66	0.54	0.85
Multiple mentions	4,326	0.69	0.55	0.91
Unseen entities	1,650	0.54	0.39	0.84

Table 3.7: SciBERT model performance on subsets of DMDD-E with instances in different categories. N represents the number of tested sequences in the related category.

token-level input. This may be attributed to the higher density of dataset mentions in section-level input, as sections are generally shorter than 512 tokens. This finding is also supported by the data presented in Table 3.4. The improved performance of section-level models may also suggest that splitting based on sections provides additional semantics that is advantageous for training when compared to splitting based on 512-token lengths, which ignores the semantic structure of the documents.

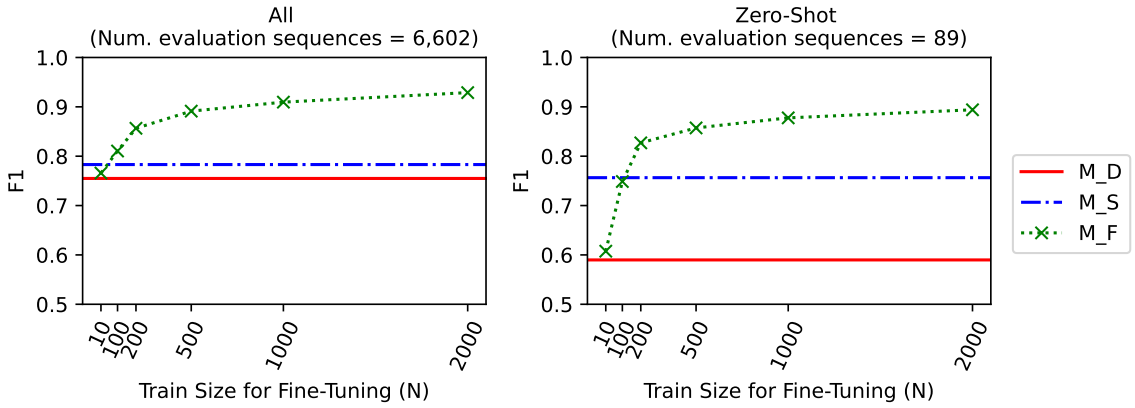


Figure 3.2: Trend of F1 when varying the number of human annotations.

3.7.1.3 Error Analysis

Based on the performance of sentence-level inputs, we conduct an error analysis on the SciBERT model and aim to identify common patterns among the erroneous

instances. As shown in Table 3.5, we observe that consistently the models have low precision and high recall, indicating a high number of false positives. After analyzing the false positives, we find that the model frequently misclassified mentions such as 'SGD' that have ambiguous meanings.

In addition, we identify three common patterns: long sequence length, multiple mentions, and unseen entities. The category of unseen entities includes not only the 10 zero-shot entities but also entities that are labeled by human annotators but cannot be linked to the DMDD dictionary. None of the unseen entities has any annotated mention in the training dataset.

Table 3.7 presents the F1, precision (P), and recall (R) of the SciBERT model on subsets of DMDD-E, grouped by the common patterns identified earlier. Performance scores are computed based on the exact match. SciBERT performed worse than average on all common patterns, with the poorest performance in the unseen category. This is consistent with the zero-shot performance presented in Table 3.5.

3.7.1.4 Fine-Tuning with Strong Labels

To evaluate the efficacy of DMDD for training purposes, we conduct a comparative analysis of SciBERT models that are trained solely with weak labels from DMDD and those trained solely with human labels from SciREX. We also examine the minimum number of human labels required to fine-tune a model for achieving a similar level of performance. We split the DMDD-E and SciREX into training sets (DMDD-E-Tr and SciREX-Tr) and testing subsets (DMDD-E-Te and SciREX-Te), where all sequences containing zero-shot entities are allocated to the testing set. We do not

train or test with negative sequences, which contain no dataset mention. This is done to investigate the effect of fine-tuning using human labels while isolating the influence from negative samples.

We developed three types of SciBERT models, as follows: (1) M_D, which is trained using DMDD; (2) M_S, which is trained using SciREX-Tr, which has 4900 manual annotated sequences; (3) M_F, which is fine-tuned on top of M_D using N sequences that are randomly sampled from DMDD-E-Tr. We conduct experiments with different N values, including 10, 100, 200, 500, 1000, and 2000.

All models are then evaluated on DMDD-E-Te. Figure 4.2 depicts the performance of the models and the F1 trend when varying the number of human annotations. The performance patterns from the overall testing set and the zero-shot subset are similar. As anticipated, the model (M_D) trained with only weak labels underperforms the models (M_S) trained with human-annotated labels. We observed that for M_F, fine-tuning with 100 strong labels enables a better performance than M_S, which is trained solely with strong labels. In other words, fine-tuning the pre-trained model from DMDD with approximately 5 human-annotated documents yields a performance similar to the model trained with around 245 human-annotated documents. Furthermore, fine-tuning with 1,000 human-annotated sequences leads to a further improvement in performance, achieving 0.9 F1 scores on DMDD-E-Te.

3.7.1.5 Train Size vs. Performance

As part of our ablation study, we investigate the training benefits resulting from the large size of DMDD. To this end, we trained SciBERT on sentence level

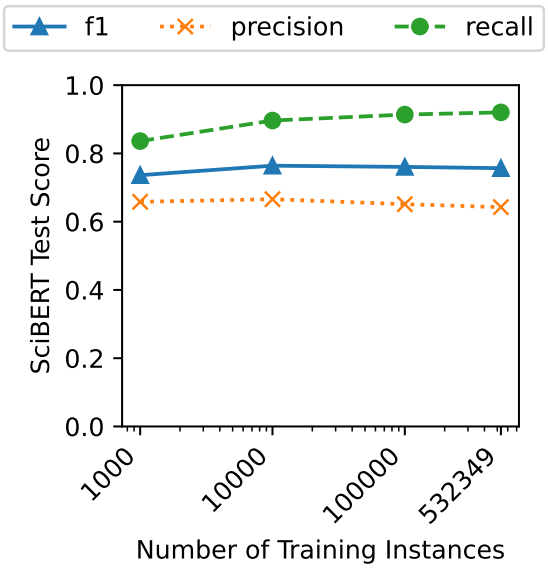


Figure 3.3: Test performance of SciBERT when training on DMDD as the train size increases.

using different sizes of DMDD, while maintaining an 80%-20% ratio between positive sequences and negative sequences. We then evaluate the trained models using DMDD-E and calculate their performance scores based on exact match. The results are presented in Figure 4.3.

Our analysis reveals that the most significant improvement in model performance occurs when increasing the training size from 1000 to 10000. The recall score continues to improve as the training size increases, while the F1 score and precision remain stable beyond a training size of 10000. This suggests that the model is predicting more false positives when the training size increases. To better leverage the large size and the diverse mentions in DMDD and enhance the model's performance, it can be beneficial to balance the training datasets before training. For instance, sampling more samples with the common features of the challenging cases discussed in Section 4.4.2 can be a fruitful strategy.

EL method	R@1	R@3	R@5	R@10	R@50
BM25	0.340	0.531	0.541	0.541	0.720
ColBERTv2	0.354	0.550	0.578	0.632	0.726

Table 3.8: Entity linking performance evaluated by recall with top K entity (R@K).

3.7.2 Entity Linking

Table 3.8 presents the experimental results for the Entity Linking (EL) task on our dataset, employing both sparse retrieval (BM25) and dense retrieval (ColBERTv2) methods. Despite not being fine-tuned, ColBERTv2 outperforms BM25, particularly in terms of R@10. However, there remains significant potential for model improvement in EL for dataset entities. For BM25, most of the errors occur due to the mentioned abbreviations that never appear in the KB. For instance, researchers may use 'H3.6M' to represent the 'Human3.6m' dataset, but this abbreviation never appears in any entity's description text in the KB. For ColBERTv2, many errors occur when the sentences with dataset mention are not descriptive of the dataset, making it difficult for the model to disambiguate based on context. An example is the sentence 'We test our method on H3.6M'.

3.8 Limitations and Future Work

The DMDD corpus is annotated through distant supervision, which prioritizes scale over accuracy. The current scope of DMDD is limited to dataset mentions that can be linked to the DMDD dictionary, resulting in missing labels for dataset mentions that are not listed on PwC websites or that have variations not included in the regular expression. This limitation may introduce annotation noise, especially when dealing

with dataset subversions that are not explicitly listed in PwC. Furthermore, DMDD does not include annotations for ambiguous cases, where distinct datasets have the same name or share acronyms, nor does it consider changes in naming conventions over time. Similar limitations apply to other corpora created using distant supervision, as annotation accuracy heavily relies on manual correction. To address these limitations, future work can focus on developing more advanced methods for mention detection and exploring alternative approaches to distant supervision. Additionally, DMDD can be extended to include annotations for more challenging test instances, such as unseen mentions, ambiguous mentions, and mentions with diverse sub-versions. In the next chapter, We propose a revised versions that have larger sizes and additional (manual) annotations for scientific entities such as model and method names.

In terms of model performance, the baseline models showed limitations when presented with unseen entities, lengthy inputs, and multiple entities. These challenges highlight the difficulties of dataset mention detection and linking in scientific literature. To develop a more robust mention detection method, future research may also explore end-to-end framework for dataset entity mention detection and linking, advanced detection networks that are robust to noise in training data, or how to leverage the context out of the mention sentence to boost the performance of EL. In addition to these approaches, future work may also explore the use of footnotes and citations in literature to improve dataset entity recognition.

3.9 Conclusion

In conclusion, DMDD is a valuable resource for studying dataset mention detection in scientific literature. As the largest corpus created for this purpose, it addresses the limitations of existing corpora in terms of size, diversity of dataset mentions, and entity linking information. Our experiments with baseline models show that DMDD enables the training of more robust models with a small number of manual labels, as demonstrated by the improved performance of SciBERT trained on DMDD compared to other corpora. The analysis of DMDD instances and experimental results highlight the challenges and open problems in the task of dataset mention detection. We believe that DMDD will stimulate further research in this important area of scientific information extraction.

CHAPTER 4

SCIDMT: A LARGE-SCALE CORPUS FOR DETECTING SCIENTIFIC MENTIONS

4.1 Introduction

Building on our prior work in dataset mention detection (Chapter 3), we now address the broader challenge of recognizing diverse scientific entities—including *methods* and *tasks*—in addition to datasets. While scientific entity mention detection (SEMD) is fundamentally a Named Entity Recognition (NER) task requiring token-level tagging, it presents unique challenges that distinguish it from general-domain NER.

The existing corpora like RCC[1], SciERC, SciREX, and TDMSci [9, 10, 11, 13, 12, 37, 35, 80, 92] have been instrumental for SEMD algorithm evaluation but are constrained by their small volume and entity linking capabilities. These limitations stem from the manual curation process, which, while ensuring quality, is resource-intensive and scales poorly.

In this paper, we present SciDMT, a corpus featuring comprehensive entity annotations spanning datasets, methods, and tasks. SciDMT contains weakly labeled instances for model training and manually annotated instances for evaluation, offering a comprehensive resource for the advancement of SEMD.

The creation of SciDMT is facilitated by distant supervision [93], leveraging document-level annotations from the Papers with Code[2] (PwC) website. This

[1]https://github.com/Coleridge-Initiative/rclc
[2]https://paperswithcode.com/

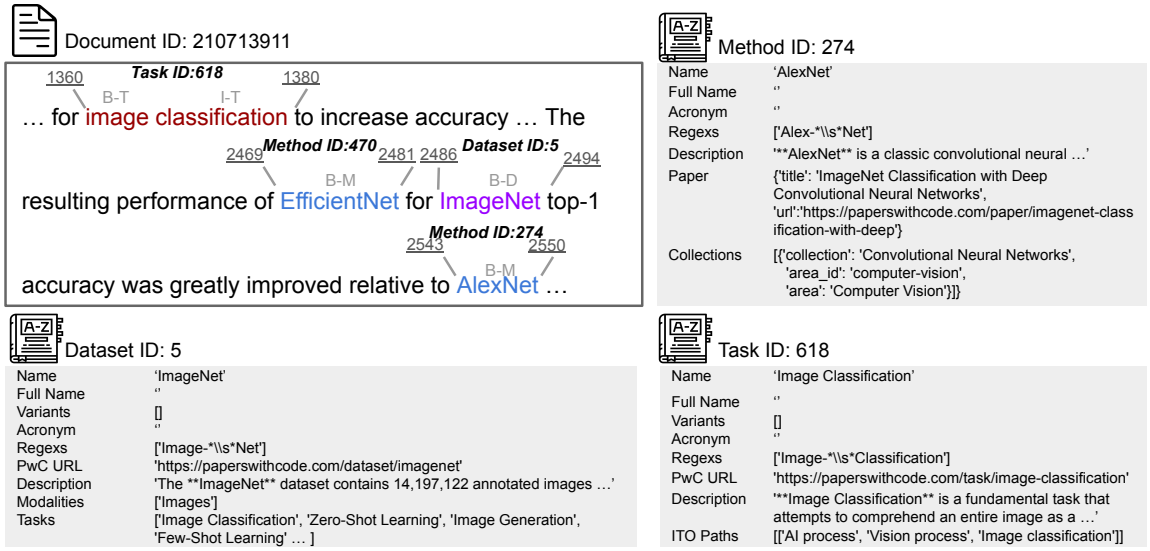


Figure 4.1: Example document-level annotation (top-left) and dictionary entries in SciDMT. We mark each occurrence of **dataset (D)**, **method (M)** and **task (T)** in papers and give the in-text spans, *entity indexes* and the BIO tags. For example, the method mention 'EfficientNet' spans from 2469 to 2481 and has a BIO tag as 'B-M'.

approach yields a main corpus comprising 48,049 machine-learning articles annotated with in-text spans, marking the mentions of datasets, methods, and tasks (DMT). Although distant supervision does not achieve the precision of manual annotations, the volume of data it generates is instrumental for training competitive models [94, 95, 96, 97, 98].

Our contributions are multifaceted. SciDMT is more than a corpus; it's a resource for enhancing information extraction. By annotating full articles and preserving the context of entity mentions, SciDMT aids in term disambiguation and enhances recognition accuracy. Every mention is linkable to PwC, and our introduction of ontology-linking for tasks and datasets further enriches the corpus's utility.

SciDMT is particularly valuable for indexing scientific papers, facilitating advanced information retrieval, and making scientific knowledge more accessible.

We validate SciDMT's efficacy through experiments, showcasing its superiority in training SEMD models compared to existing corpora. Furthermore, our evaluation of NER methods, including SciBERT and GPT-3.5, on SciDMT demonstrates the intricate challenges and prospects of SEMD. The SciDMT corpus can be accessed at HuggingFace Hub[3]. Our contributions can be summarized as follows:

- We introduce SciDMT, a SEMD corpus annotated at the document level, covering datasets, methods, and tasks. Each mention is linked to PwC and enriched with ontology-linking, offering a comprehensive resource for information extraction.
- We compare SciDMT to existing corpora and demonstrate its effectiveness in training competitive SEMD models.
- We evaluate several NER methods, including SciBERT and GPT-3.5, on SciDMT, and discuss the unique challenges encountered in SEMD.

4.2 SciDMT Corpus

In this section, we describe the construction of SciDMT's main corpus and the human-annotated evaluation sets. We also present a comprehensive comparison between SciDMT and related corpora.

[3]https://huggingface.co/datasets/jopan/SciDMT

4.2.1 SciDMT's Main Corpus

We present the construction of our primary corpus in this subsection. Figure 4.1 is an illustrative example of a SciDMT data entry, which includes the parsed scientific article and the in-text annotation for scientific mentions.

4.2.1.1 Data Collection

Although our data collection methodology is similar to the one in DMDD [99] in that parsed articles from S2ORC [78] and document-level annotations from Papers With Code (PwC) are utilized, we significantly extend their distance supervision annotation. We extract publications' metadata of methods and tasks from PwC and dataset information directly from the paper's PwC webpage. This process yields 48,049 matched papers between S2ORC and PwC, identified via their ArXiv IDs.

4.2.1.2 Annotation with Distant Supervision

Utilizing user-provided data from PwC, we aimed to align these entity names with their occurrences in the body of the articles. Strict matching, though generally effective, encounters challenges due to the occasional inconsistencies in entity naming conventions between PwC and authors (e.g., 'k-Means' vs. 'k-Means Clustering', 'GoogLeNet' vs. 'GoogleNet').

To address this, we developed a comprehensive DMT entity dictionary with regular expressions (regex), which enables us to accommodate variations in entity naming with approximate matching. These regex are not crafted for individual entities but are

generated based on a universal set of rules, enhancing the scalability of our annotation process.

The regex creation rules can be summarized as follows. First, optional spaces and dashes are allowed between words. Second, acronyms are created for entity names with multiple words, but only when the original entity name is not already an acronym. Third, various common version names are considered. For example, if 'v3.0' appears in the name, we allow matching mentions with 'v3', and if '18' appears in the name, we allow matching mentions with '2018'. Fourth, verbs in different tenses and nouns in plural and singular forms are allowed. Lastly, the casing is ignored in regex, except for special cases such as the lowercase of the name being a common English word or the name being very short. As such, each entity name variation has one regular expression. Examples of regex can be seen in Figure 4.1.

Our annotation process, though not aiming for optimal regex creation, is designed to obtain a substantial volume of weakly labeled data, instrumental for the effective training of NER models.

To enhance the entity linking capabilities of SciDMT, we integrated ontology paths from ITO [100], which offers a structured hierarchy of AI tasks and datasets. In this integration, we mapped task and dataset entities from SciDMT to their corresponding elements within the ITO hierarchy. For task entities, we showed the complete hierarchy path in ITO, whereas for dataset entities, we showed the associated tasks. This method established relationships between entities reflecting their positions in the ontology's structure. For instance, datasets used in'Image Classification' tasks are linked together, and tasks like 'Image Classification' and'Image Segmentation' are

connected as they both fall under the category of'Vision Process'. This integration of ontology paths not only enhances the comprehensiveness of SciDMT but also enriches its usages for detailed entity analysis.

All 48,049 articles are annotated with the full text, section spans, and both document-level and in-text entity annotations. The annotations are indexed to the SciDMT entity dictionary, as illustrated in Figure 4.1.

4.2.1.3 Data Preprocessing

In this phase, we employ a comprehensive approach, combining all regular expressions crafted from the SciDMT's dictionary. This exhaustive search is applied across all 48,049 articles, aiming to capture and annotate a broader spectrum of DMT entity mentions, thereby mitigating the issue of missing mentions.

	Inst.		Dataset		Task		Method		All	
Corpus	**Unit**	**# Inst.**	**# M.**	**# U.M.**	**# M.**	**# U.M.**	**# M.**	**# U.M.**	**# M.**	**# U.M.**
SciERC	abstract	500	770	644	1,281	1,067	2,090	1,760	4,141	3,445
SciREX	paper	438	10,615	2,865	32,526	12,893	98,458	34,030	141,599	47,974
TDMSci	sentence	444	612	478	1,615	999	0	0	2,227	1,476
SciDMT	paper	48,049	449,798	10,807	647,360	7,850	733,728	16,579	1,830,886	34,648

Table 4.1: Summary of corpora for scientific entities mention detection.

4.2.2 Evaluation Sets with Human Annotations

We manually annotated two sets of instances for evaluation purposes, one from SciDMT and the other from SciREX. The inclusion of SciREX serves a dual purpose: it not only facilitates a comparative analysis of dataset quality but also aids in assessing the complexity level of our dataset during experimental evaluations. These evaluation sets were manually annotated by two NLP researchers using brat rapid

annotation tool [81]. We aggregated the annotations by keeping only the mentions where both annotators agreed.

For SciDMT evaluation set called **SciDMT-E**, annotators were tasked with 100 papers that were sampled with the most number of DMT mentions among the randomly sampled SciDMT's valid set. Additionally, we randomly selected 10 unseen entities from each DMT category and annotated the 256 sentences containing these unseen entities. Annotators were instructed to verify the detected mentions from SciDMT's main corpus and identify any missing mentions in each paper. To ensure accuracy, annotators were directed to search the PwC website and Google to confirm the DMT entities during the annotation process. Full annotation instructions are provided in HuggingFace Hub[4].

We assessed the level of agreement between annotators using the relaxed span matches method, which considers a match when the entity mention spans from the annotators overlap. On SciDMT-E, the resulting Cohen Kappa 0.87 represents a substantial inter-annotator agreement [101]. SciDMT-E contains 14,846 sentences with DMT entities, where 3,345 sentences contain dataset mentions, 11,124 sentences contain method mentions, and 5,899 sentences contain task mentions. The annotated mentions in SciDMT-E can be linked to 1,070 entities listed in SciDMT's dictionary. On average, each annotator required approximately 1 minute to annotate one sentence or 30 minutes to annotate one document.

When using SciDMT-E as ground truth and exact match for comparison, SciDMT's weak annotation obtains an F1 score of 61.9%, precision of 50.8%, and

[4]https://huggingface.co/datasets/jopan/SciDMT/re solve/main/SciDMT%20Annotation%20Guideline.pdf

recall of 79.4%. The recall rate indicates that the distantly-implied signals from PwC are able to capture 79.4% of scientific entities in text. The low precision suggests that a significant portion of human-annotated mentions does not exactly match with the machine-annotated mentions. This observation is attributed to many weak labels failing to include the full entity name. For example, distant supervision may provide a partial annotation in sentences containing 'KITTI 2012', tagging 'KITTI' but ignoring the version part of the name, i.e., '2012'.

For SciREX evaluation set (SciREX-E), we used the same annotation guideline to annotate 10 papers for DMT entities. The Cohen Kappa 0.76 also indicates a substantial agreement between annotators. SciREX-E contains 2,207 sentences with DMT entities. Compared to SciREX-E, SciREX's original annotation obtains an F1 score of 65.4%, precision of 77.4%, and recall of 56.6%. The low recall suggests that many mentions were missing in the original annotations. For example, the original annotation often misses the 'ImageNet' mentions in phrases such as 'ImageNet pre-trained model'.

4.2.3 Comparison with Related Corpora

We compare SciDMT with three related corpora in terms of size (Table 4.1) and quality. For each of the three scientific entity types, we give the total number of entity mentions (# M.) and the total number of unique mentions (# U.M.) for each corpus.

4.2.3.1 Corpora Size

SciDMT is larger than the related corpora, in terms of the number of instances (# Inst. = 48,049), instance units (Inst. Unit = Paper), and the number of mentions (# M. of All = 1,830,886). Having document-level annotations compared to sentence-level annotations, SciDMT allows a larger model input scope (e.g., sentence before and after the target sentence), allowing for richer contextual information. Furthermore, since entity mentions in SciDMT appear in multiple sentences across the 48K papers, it provides a diverse set of training data for NER and Entity Linking. This is captured by comparing # M. and # U.M. in Table 4.1. A large number of unique mentions indicates a wide range of scientific entities captured in SciDMT, while the high total number of mentions contributes to the training of robust models by providing a variety of background semantics related to scientific entities.

4.2.3.2 Entity Linking Annotation

Entity linking is the task of associating mentions in the text with their corresponding entities in knowledge bases, such as Wikipedia and Papers with Code. In the case of scientific entity mentions, entity linking is crucial as it allows users to access the correct dataset, source code, and source papers for empirical studies. Since SciDMT is created based on Papers with Code, all entities mentioned in SciDMT have explicit links to the Papers with Code website and a unique identifier. Because of the incorporation of ITO paths, dataset and task entities have intra-entity annotation as well.

In contrast, the related corpora do not have linking information about their entities because the annotators were not instructed to provide the linking annotation. Our attempts to link the entities in the related corpora to knowledge bases, such as Papers with Code and the ACL Anthology, were largely unsuccessful due to several reasons:

1) Their mentions include extra characters or text strings. For example, the mention 'fine-tuned U-Net' includes the descriptive text 'fine-tuned' for the method.

2) Their mentions include more than one entity, for example, 'ImageNet pretrained VGG-19'.

3) Their mentions do not include the actual entity name, for example, 'models' and 'methods'. This is because some related corpora are annotated with pronominal reference to entities, as defined in ACE 2005 [82]. Pronominal reference is not helpful in linking mentions to scientific entities, especially when the corpora are not annotated on the paper level and the proper name reference is missing from the annotated instance. Within this characteristic group, there are also confusing mentions not using the most commonly used names or missing parts of the names. For example, 'VGG' can denote many possible models, such as VGG-16 and VGG-19. This further points toward the value of including linking attributes in the annotation whenever possible, as done in our work.

4.3 Experimental Setup

The experiments are designed for the task of scientific entity mention detection (SEMD) with three primary objectives in mind: establishing baseline performance on SciDMT, gaining insights into the difficulty of SEMD, and evaluating the effectiveness

of using SciDMT for training. The experiments focus on three categories of scientific entities: datasets, methods, and tasks (DMT).

4.3.1 Baseline Models

We formulate the task of SEMD as a single-sentence tagging task, and we include a diverse set of models as baselines in our evaluation, namely Conditional Random Fields (CRF), Bidirectional Long Short-Term Memory (BiLSTM), BERT [85], SciBERT [86] and GPT-3.5 [102].

For CRF, BiLSTM, BERT, and SciBERT, we conduct training in 3 rounds using randomly shuffled training sets. For CRF, we incorporate features such as Part-of-Speech (POS) tags and keywords.

For BiLSTM, we evaluate two additional variations where the pre-trained embedding layer is initialized with either GLoVe (BiLSTM-G) [87] or Word2Vec (BiLSTM-W) [88]. Tokens that are not mapped with pre-trained embeddings are initialized with zeros. The embedding layer for all tokens is updated during training. To ensure a fair comparison, we set the embedding dimension to 300 for BiLSTM, BiLSTM-G, and BiLSTM-W. For BiLSTM-G, we utilize the embedding trained on Wikipedia and Gigaword, covering 30,612 tokens, while 134,802 tokens are missing in the pre-trained embeddings. For BiLSTM-W, we use the embedding trained on Google News and convert 68,553 tokens, with 96,861 tokens missing. We notice that many scientific entity names are missing in the pre-trained embeddings.

For BERT and SciBERT, we use the pre-trained weights of base-cased BERT [85] and scivocab-cased SciBERT [86]. We keep the same hyperparameters for training

Subset	SciREX-E: All			SciDMT-E: All			SciDMT-E: Unseen		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
CRF	.270 ± .000	.172 ± .000	.626 ± .000	.455 ± .000	.313 ± .000	.832 ± .000	.185 ± .000	.107 ± .000	.683 ± .000
BiLSTM	.328 ± .004	.238 ± .003	.525 ± .007	.520 ± .002	.411 ± .003	.708 ± .002	.176 ± .017	.137 ± .009	.251 ± .048
BiLSTM-G	.325 ± .007	.235 ± .007	.527 ± .005	.526 ± .005	.414 ± .007	.721 ± .005	.192 ± .039	.147 ± .029	.279 ± .060
BiLSTM-W	.329 ± .003	.238 ± .004	.529 ± .002	.523 ± .004	.411 ± .006	.719 ± .002	.188 ± .023	.137 ± .016	.304 ± .045
BERT	.480 ± .007	.372 ± .010	**.674 ± .005**	.643 ± .004	.523 ± .007	**.835 ± .006**	.747 ± .007	.721 ± .011	.776 ± .008
SciBERT	.490 ± .003	.388 ± .004	.666 ± .006	**.649 ± .001**	**.531 ± .002**	.833 ± .003	**.763 ± .009**	**.737 ± .012**	**.792 ± .023**
GPT-3.5	**.503 ± .000**	**.499 ± .000**	.506 ± .000	.586 ± .000	.672 ± .000	.520 ± .000	.484 ± .000	.701 ± .000	.370 ± .000
Subset	SciDMT-E: Datasets			SciDMT-E: Methods			SciDMT-E: Tasks		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
CRF	.590 ± .000	.449 ± .000	.858 ± .000	.410 ± .000	.276 ± .000	.799 ± .000	.393 ± .000	.259 ± .000	.813 ± .000
BiLSTM	.551 ± .002	.438 ± .004	.743 ± .004	.474 ± .003	.363 ± .003	.684 ± .001	.489 ± .002	.377 ± .003	.696 ± .003
BiLSTM-G	.558 ± .005	.443 ± .007	.756 ± .004	.480 ± .007	.365 ± .008	.698 ± .003	.496 ± .004	.382 ± .006	.706 ± .006
BiLSTM-W	.552 ± .005	.435 ± .005	.755 ± .007	.476 ± .005	.363 ± .006	.693 ± .000	.492 ± .007	.377 ± .008	.708 ± .001
BERT	.679 ± .004	.550 ± .005	**.886 ± .003**	.602 ± .005	.478 ± .008	.812 ± .004	.560 ± .001	.430 ± .005	**.804 ± .013**
SciBERT	**.678 ± .003**	**.551 ± .002**	.881 ± .003	**.611 ± .001**	.490 ± .001	**.813 ± .003**	.565 ± .003	.438 ± .004	.795 ± .007
GPT-3.5	.663 ± .000	.729 ± .000	.608 ± .000	.582 ± .000	**.628 ± .000**	.543 ± .000	**.579 ± .000**	**.620 ± .000**	.543 ± .000

Table 4.2: NER model performance on human-annotated evaluation sets. In each column, the highest score is shown in **boldface**.

the models as in the original SciBERT [86], except for the batch size, which is set to 16.

GPT-3.5 is included in our model selection because Large Language Models (LLMs) have demonstrated impressive natural language understanding capabilities, including the capability for entity recognition [103]. We use Spacy-LLM [5], which is a Python package that combines the language processing library spaCy with LLM backends. In terms of model specifications, we use spacy.NER.v2 as task, 'DATASET, METHOD, TASK' as labels, and OpenAI's gpt-3.5-turbo-0613 as the LLM backend. The input to the model consists solely of the sentence text. The model's output is in the span format, which we convert to token-level BIO labels for evaluation purposes. We only use the Spacy-LLM's zero-shot setting without any examples or label definitions. We acknowledge that more sophisticated model tuning and prompt engineering may yield improved performance; however, our focus here is on presenting baseline results.

[5]https://github.com/explosion/spacy-llm

GT:	One of the most prominent models of this sort is the **Feature Pyramid Network** (**FPN**) proposed by Lin et al.
SciBERT:	One of the most prominent models of this sort is the **Feature Pyramid Network** (**FPN**) proposed by Lin et al.
GPT-3.5:	One of the most prominent models of this sort is the **Feature Pyramid Network (FPN)** proposed by **Lin et al.**
GT:	Note, that the concatenation mode is only relevant for training the **softmax** classifier.
SciBERT:	Note, that the concatenation mode is only relevant for training the **softmax** classifier.
GPT-3.5:	Note, that the concatenation mode is only relevant for **training** the **softmax classifier**.
GT:	To achieve such progress, we consider that **Kinetics** for **3D CNNs** should be as large-scale as **Ima-geNet** for **2D CNNs**, though no previous work has examined enough about the scale of **Kinetics**.
SciBERT:	To achieve such progress, we consider that Kinetics for 3D CNNs should be as large-scale as Ima-geNet for 2D **CNNs**, though no previous work has examined enough about the scale of Kinetics.
GPT-3.5:	To achieve such progress, we consider that **Kinetics** for **3D CNNs** should be as large-scale as Ima-geNet for **2D CNNs**, though **no previous work** has examined enough about the scale of **Kinetics**.
GT:	Using **biLMs** for supervised NLP tasks Given a pre-trained **biLM** and a supervised architecture for a target NLP task, it is a simple process to use the **biLM** to improve the task model.
SciBERT:	Using biLMs for supervised NLP tasks Given a pre-trained biLM and a supervised architecture for a target NLP task, it is a simple process to use the **biLM** to improve the task model.
GPT-3.5:	Using biLMs for **supervised** NLP tasks Given a pre-trained biLM and a **supervised** architecture for a **target NLP task**, it is a simple process to use the biLM to improve the task model.

Table 4.3: Prediction examples for SciBERT and GPT-3.5 on evaluation samples from SciDMT-E. Where the predicted mention tokens are highlighted for **dataset (D)**, **method (M)** and **task (T)**.

4.3.2 Train-Valid Split

To establish a train-valid split for SciDMT's main corpus and SciREX, we first perform a random document-level split. Next, we randomly select 30 scientific entities, with 10 entities chosen from each DMT category in SciDMT. These entities form the unseen set, which is exclusively included in the valid set and the evaluation set, and is excluded from the training set of any corpus. Finally, we conduct a sentence-level train-valid split based on the aforementioned document-level split.

At the document level, the train set of SciDMT consists of 36,635 documents (76%), while the valid set comprises 11,414 documents (24%). At the sentence level, the train set of SciDMT contains 738,857 positive sentences (70%) that contain mentions of DMT entities, while the valid set consists of 314,689 positive sentences (30%).

4.4 Experimental Results

In this section, we present the experimental specifications and report the results of our experiments. The evaluation is conducted on the manually annotated evaluation sets: SciDMT-E and SciREX-E. The average and standard deviation of performance scores, including F_1, precision (P), and recall (R), are calculated based on the exact scores.

4.4.1 Baselines Evaluation

The NER models discussed in Section 4.3.1, undergo 3 rounds of training using randomly shuffled training sets from SciDMT, except GPT-3.5, which is evaluated for 1 round. Performance scores are computed on SciDMT-E, SciREX-E, and various subsets of SciDMT-E. The results are summarized in Table 4.2 and prediction samples are shown in Table 4.3.

The performance on the two evaluation sets, SciREX-E and SciDMT-E, is similar, indicating comparable dataset difficulty. Surprisingly, for BERT and SciBERT, the performance of the unseen subset in SciDMT-E is higher than the overall average performance, contrary to our expectations. This may be due to the limited sample size, which might have excluded more challenging cases. Additionally, we observe that dataset mentions are generally easier to detect compared to method and task mentions, possibly because dataset names are more standardized and have fewer naming variations.

SciBERT and BERT exhibit similar performances among the different models, achieving the highest overall performance. The variations of BiLSTM using Word2Vec embedding (BiLSTM-W) and GloVe embedding (BiLSTM-G) perform similarly to the original BiLSTM, without notable improvements. This aligns with previous research that has shown Word2Vec and GloVe to be equivalent in terms of module hyperparameter tuning [104, 105]. Without sophisticated feature learning, CRF does not perform as competitively as the other models.

GPT-3.5 without fine-tuning achieves slightly lower scores compared to the trained models, but still demonstrates knowledge about scientific entities without explicit learning. It predicts some of the general concept words (e.g.: 'training', 'inference', and 'modification') and citations (e.g.: 'Fortunato et al. 2017') as scientific entities, which are not included in our manual evaluation sets. We hypothesize that as GPT-3.5 is trained with Common Crawl and scientific papers often have web presences, GPT-3.5 may have read many scientific papers during training and accumulated knowledge about understanding and identifying scientific entities. By including SciDMT in its training or employing few-shot learning, the performance of GPT-3.5 can potentially be further improved. However, GPT-3.5 struggles with isolating the correct entity from descriptions or strings with multiple mentions, as shown in Table 4.3. Additionally, like other trained models, GPT-3.5 encounters difficulties in recognizing mentions with uncommon dash patterns, such as 'Ima-geNet'.

Eval. Group	N	F1	P	R
All	17,053	.649	.531	.833
Long Sequences	1,670	.547	.420	.786
Multiple Mentions	8,312	.577	.440	.836
Unseen Mentions	4,212	.529	.423	.708

Table 4.4: Error analysis for SciBERT. N represents the number of evaluated sequences with different features.

4.4.2 Error Analysis

Based on the performance of the best model, SciBERT, we conduct an error analysis to identify common patterns among erroneous instances. These patterns include long sequences with more than 200 characters, sequences with multiple mentions, and sequences with unseen mentions.

Here,'unseen mentions' are twofold: firstly, they include the annotated unseen entity mentions previously discussed. Secondly, they encompass mentions identified by human annotators that could not be linked to the SciDMT dictionary. Like the annotated unseen entities, these unseen mentions lack any representation in the training dataset.

We compute the number of evaluated sentences exhibiting each pattern and their corresponding performance scores in Table 4.4. SciBERT demonstrates below-average performance across all common patterns, with the lowest performance observed in the unseen category.

In Table 4.3, the last two examples are samples demonstrate cases of long sequences and multiple mentions, while the last example is an sample for unseen mentions as it is the one containing unseen entity 'biLM'.

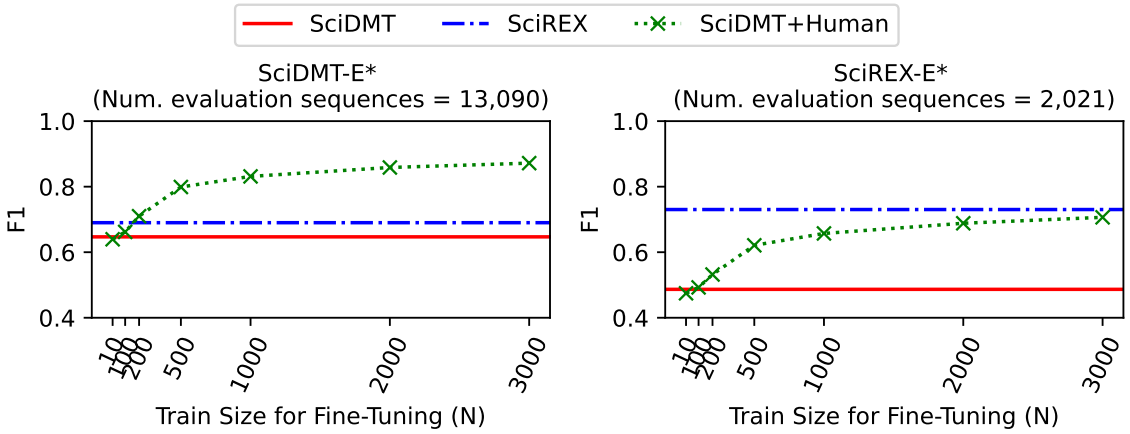


Figure 4.2: Trend of F1 when varying the number (N) of human-annotated samples used for fine-tuning. Each line in the graph, represented in the legend, corresponds to a model being trained with a distinct dataset.

4.4.3 Fine-Tuning with Human Labels

To assess the effectiveness of SciDMT as a training resource, we compare SciBERT models trained solely with weak labels from SciDMT to those trained solely with human-annotated labels (human labels) from SciREX. We also investigate the minimum number of human labels needed to achieve a comparable level of performance. For the fine-tuning training set, we randomly sample 1500 positive sequences from each of our human-annotated evaluation sets: SciDMT-E and SciREX-E, while retaining the remaining sequences as the fine-tune evaluation sets (SciDMT-E* and SciREX-E*).

We develop three types of SciBERT models:

- M_{SciDMT}, which is trained using the weak labels from SciDMT.
- M_{SciREX}, which is trained using the human labels from SciREX, comprising only human-annotated samples. The training set of SciREX consists of 60,021

positive sequences, excluding those that overlap with SciDMT's valid set and our human-annotated evaluation sets.

- $M_{SciDMT+N}$, which is fine-tuned on top of M_{SciDMT} with N randomly sampled human labels from the fine-tuning training set. We experiment with different values of N, including 10, 100, 200, 500, 1000, 2000, and 3000.

All models are evaluated separately on SciDMT-E* and SciREX-E*. The models' performance and the trend in F1 scores as the number of human annotations varies are shown in Figure 4.2.

As anticipated, M_{SciDMT} trained solely with weak labels performs lower M_{SciREX} trained with human labels. In terms of fine-tuning, $M_{SciDMT+N}$ achieves better performance than M_{SciDMT} with 100 human labels.

On SciDMT-E*, $M_{SciDMT+N}$ surpasses the performance of M_{SciREX} with only 200 human labels, outperforming the model trained with 60K human labels. Moreover, fine-tuning with 3,000 human-annotated sequences further improves the performance, achieving 0.88 F1 scores on SciDMT-E*.

On SciREX-E*, where M_{SciREX} has the advantage of being trained in the same domain, $M_{SciDMT+N}$ needs 3000 human labels to achieve similar performance to M_{SciREX}. In other words, fine-tuning the pre-trained model from SciDMT with approximately 10 human-annotated documents yields comparable performance to the model trained with around 245 documents.

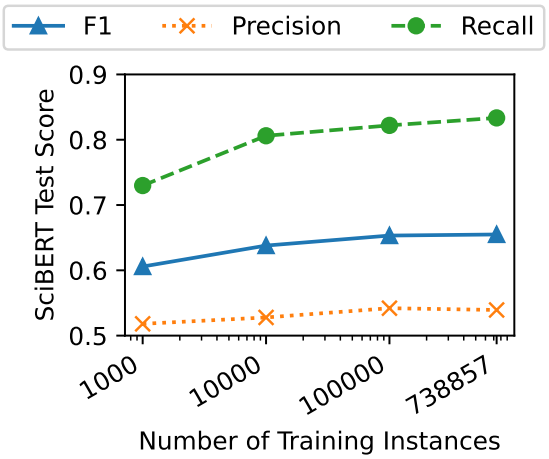


Figure 4.3: Validation performance of SciBERT when training on SciDMT as the train size increases.

4.4.4 Impact of Training Scale on Performance

As part of our ablation study, we investigate the training benefits derived from the large size of SciDMT. We train SciBERT using different training set sizes, randomly sampled from the entire training data: 10^3, 10^4, 10^5, and the complete training data consisting of 739K samples. The performance scores on SciDMT-E are plotted using a logarithmic scale in Figure 4.3.

Our analysis reveals that the most significant improvement in model performance occurs when increasing the training size from 1,000 to 10,000 sequences. The recall score continues to improve as the training size increases, while the F1 score and precision remain relatively stable beyond a training size of 100,000. This suggests that the model tends to predict more false positives with larger training sizes.

To better leverage the large size and diversity of mentions in SciDMT and further enhance the model's performance, it can be beneficial to balance the training datasets

by sampling biased toward challenging cases. This strategy can focus on samples with common features observed in the error analysis (Section 4.4.2).

4.5 Limitations

SciDMT is a large-scale corpus annotated through distant supervision. This approach sacrifices accuracy for scale. The current scope of SciDMT is limited to scientific mentions that can be linked to the SciDMT dictionary, resulting in missing labels for scientific mentions that are not listed on PwC websites or that have variations not included in the regular expression. This limitation may introduce annotation noise, especially when dealing with subversions that are not explicitly listed in PwC. In addition, SciDMT may inadvertently inherit biases from its primary source, PwC's data. This reliance could lead to disproportionate emphasis or neglect of certain topics within the corpus.

Furthermore, SciDMT does not include annotations for ambiguous cases, where distinct entities have the same name or share acronyms, nor does it consider changes in naming conventions over time. Similar limitations apply to other corpora created using distant supervision, as annotation accuracy heavily relies on manual correction.

Additionally, SciDMT does not annotate pronominal references to entities, resulting in incomplete coreference information compared to corpora like SciERC. Despite these limitations, the large-scale data obtained through distant supervision proves valuable for training deep learning models, a sentiment echoed in previous studies [94, 106, 96] and our experimental findings in Section 4.4.4.

4.6 Conclusion

We presented SciDMT, the largest corpus specifically created for the study of scientific entity mention detection (SEMD). SciDMT offers a substantial size, diverse entity mentions, and comprehensive entity-linking information, making it a valuable resource and a benchmark for the development and evaluation of advanced scientific information extraction models.

The experiments conducted using various NER models on SciDMT provide valuable insights and performance baselines for SEMD. The error analysis conducted sheds light on the existing challenges and unveils opportunities for innovation in SEMD.

Moving forward, our focus is on the iterative enhancement of SciDMT. We aim to augment the corpus by broadening the spectrum of annotated entities, refining weak labels, and increasing the corpus size. The incorporation of sophisticated post-processing techniques [107, 108, 109, 110] to cleanse distant supervision labels is also on our agenda. Additionally, future work can focus on addressing more challenging instances, such as unseen and ambiguous mentions, to further enhance the performance of scientific mention detection models.

In conclusion, SciDMT presents a significant contribution to the field of SEMD by providing a large-scale corpus and performance baselines for SEMD models. We hope SciDMT will inspire and drive future research in scientific information extraction.

CHAPTER 5

TAXONOMY-DRIVEN KNOWLEDGE GRAPH CONSTRUCTION FOR DOMAIN-SPECIFIC SCIENTIFIC APPLICATION

5.1 Introduction

Effective management and utilization of structured knowledge is a core challenge in domain-specific research. While scientific publications across fields, from materials science to epidemiology, routinely describe critical relationships between models, observational datasets, and analytical findings, these connections are rarely formalized or linked to standardized data sources. For instance, climate science papers might detail how green house gas emission affects the occurrence of wildfires [111], while chemistry studies could analyzes battery chemistry performance under different extreme conditions [112]. Yet in both cases, these insights remain trapped in unstructured text, inaccessible to computational analysis. This lack of systematization impedes cross-study knowledge integration, slowing discovery and limiting reproducibility. Knowledge graphs (KGs) address this gap by structuring entities and relationships into semantically interconnected frameworks, enabling querying, automated reasoning, and cross-domain interoperability [113].

Although KGs have advanced research in domains like material science [114] and geospatial sciences [115], constructing them in specialized fields faces two main challenges. First, existing methods overlook domain taxonomies, which are curated

hierarchies of verified entities and relationships. Instead, they build KGs from scratch via LLMs. [116]. While flexible, this forfeits the semantic rigor and community consensus embedded in taxonomies, leading to inconsistent representations. Second, despite LLMs' proficiency in general-purpose information extraction [117], they struggle in specialized domains: hallucinating entities, misclassifying relationships, and overlooking tail-domain concepts absent from their training data [118]. For example, in climate science, models frequently conflate teleconnections (large-scale climate linkages) with generic correlations or fail to recognize emerging terms like 'Arctic amplification'. These errors compromises KG reliability for downstream tasks.

A critical bottleneck in KG construction lies in accurate named entity recognition (NER) for specialized domains. State-of-the-art generalist models like GLiNER [119], which achieve competitive performance on broad-coverage benchmarks (F1: 0.478), falter in domain-specific settings—scoring only 0.339 F1 on climate science texts. This performance gap stems from two interrelated issues: 1) Domain-specific terminology—such as teleconnections, oceanic Rossby waves, and CMIP6 emission scenarios—occupies the "long tail" of knowledge underrepresented in LLM training corpora [118], and 2) LLMs lack mechanisms to disambiguate domain-relevant entities (e.g., "water" as a model variable in hydrological studies) from semantically similar generic terms (e.g., generic mentions of "water" in non-technical contexts or "signal processing" in electronics). Consequently, LLMs either omit critical concepts or misclassify them, propagating errors into downstream KG components.

To address these challenges, we propose a framework that synergizes domain taxonomies, constrained LLM extraction, and iterative validation, demonstrated

through climate science KG construction. Our approach comprises three key components: **1) Taxonomy-driven KG construction**: Extraction is anchored to expert-curated taxonomies (e.g., MeSH in biomedicine, NASA's GCMD [120] in climate science). By integrating RAG with LLMs, we ensure extracted entities (e.g., CMIP6 experiments) and relationships (e.g., ENSO influences Drought) align with the taxonomy's hierarchical structure, preserving semantic consistency. **2) Constrained Entity and Relation Typing**: To reduce hallucinations, we restrict the types of named entities (NEs) and relations that LLMs can extract. This prevents irrelevant entity types, such as person names, from being included. Few-shot learning is employed to adapt the model to domain tasks, improving performance. **3) RAG-based output verification**: Unlike approaches like GraphRAG [116], which directly use model outputs for KG construction, we verify outputs using RAG against the domain taxonomy. This prevents the introduction of wrong entities and relations into the graph.

Using climate science as our proving ground in Chapter 6, we demonstrate how this approach resolves the precision-recall tradeoffs inherent to open-domain IE systems while maintaining computational tractability. Our work advances domain-specific KG construction through the following contributions:

- **A Generalizable Taxonomy-Driven Methodology**: While demonstrated in climate science, our framework provides a blueprint for constructing KGs in any domain with structured taxonomies (e.g., Space Domain Awareness taxonomy). By anchoring extraction to expert-curated hierarchies, we ensure semantic consistency while enabling sustainable updates.

- **Hallucination-Robust LLM-RAG Integration**: We demonstrate how RAG-enhanced LLMs, constrained by taxonomic rules, reduce entity hallucination by 23% compared to baseline methods while maintaining 47% recall on tail-domain concepts.
- **Rigorous Evaluation Framework**: Ablation studies and cross-model comparisons quantify the impact of taxonomy anchoring, showing 18% F1 gains over SOTA models like GLiNER in climate science NER—a pattern generalizable to other specialized domains.

This work bridges unstructured scientific text and structured knowledge representation, offering a scalable solution not only for climate science but for any domain requiring precise, taxonomy-grounded KGs. By addressing the dual challenges of semantic consistency and domain adaptability, our framework empowers researchers to systematically organize evolving knowledge while preserving interoperability with established taxonomies.

5.2 Method Overview

We propose a generalizable framework for constructing domain-specific KGs that harmonizes structured taxonomies with unstructured text extraction. While demonstrated through climate science, a domain with complex terminology and rapid conceptual evolution—the methodology applies to any field with curated vocabularies (e.g., Unified Astronomy Thesaurus or GeoNames in geospatial sciences). The framework comprises three stages: **1) Taxonomy as Semantic Scaffold**: Domain taxonomies (e.g., GCMD for climate science) define entity hierarchies and

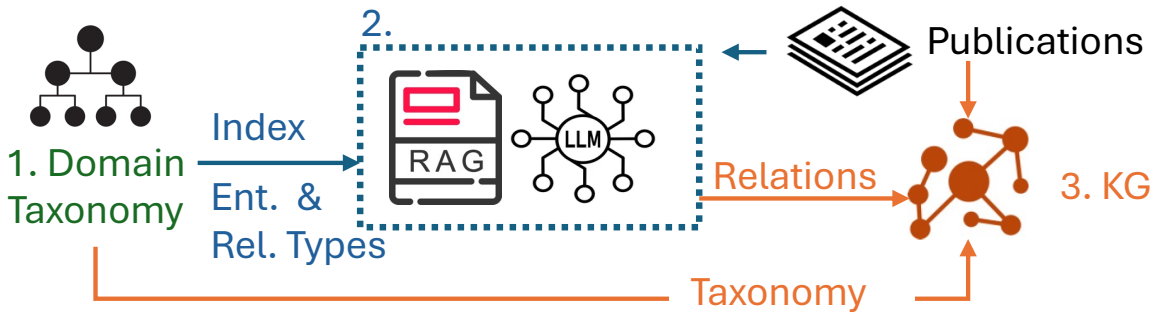


Figure 5.1: Overview of the proposed framework for Knowledge Graph construction

relationship rules, ensuring consistency. **2) LLM-RAG Hybrid Extraction**: RAG grounds LLMs in taxonomy entities during extraction, reducing hallucinations while preserving contextual nuance. **3) Dynamic KG Assembly**: Validated entities and relationships are integrated into a graph that evolves with publications, balancing taxonomic rigor with conceptual growth.

Figure 5.1 illustrates the proposed framework for KG construction from scientific publications. We start with a taxonomy, which provides a hierarchical classification of domain-specific named entities but lacks explicit relationships beyond hierarchical structures such as subclass relations. To enrich this taxonomy, we incorporate a broader set of relations that define interactions between entities. These relations are automatically derived from research publications, but are constrained by our RAG to predefined types of relations and entities within the taxonomy, ensuring consistency and mitigating hallucinations. The taxonomy serves as the structural foundation of the KG, anchoring entity organization, while the extracted relations add depth by capturing meaningful interactions between entities.

5.3 Stage 1: Taxonomy Integration

We propose a 3-step framework to transform domain taxonomies into adaptive backbones for KG construction, applicable to scientific fields requiring structured yet evolving knowledge representation. Using climate science as a case study, the process involves: aggregating domain-specific taxonomies, enhancing node definitions, and indexing for semantic alignment. This is detailed in Section 6.2.

All entities are embedded using NVIDIA NV-Embed-v2 [121] (4096 dimensions), a top-performing model on the MTEB benchmark [122]. The embeddings enable semantic search and link literature-extracted knowledge to taxonomy. This indexing ensures the taxonomy serves as a stable anchor for maintaining semantic consistency across the evolving KG.

5.4 Stage 2: Information Extraction via LLM-RAG Synergy

Figure 5.2 outlines our 3-step pipeline for taxonomy-guided information extraction: 1) prompt engineering, 2) constrained entity/relationship extraction, and 3) validation against domain taxonomies. Below we detail each stage.

5.4.1 LLM Prompt Construction

A trivial prompt asking the LLM to extract entities and relationships from domain science literature is insufficient for ensuring accuracy, consistency, and alignment with domain knowledge. Without constraints, the model tends to hallucinate entity types, introduce ambiguous relationships, and deviate from the standardized terminology

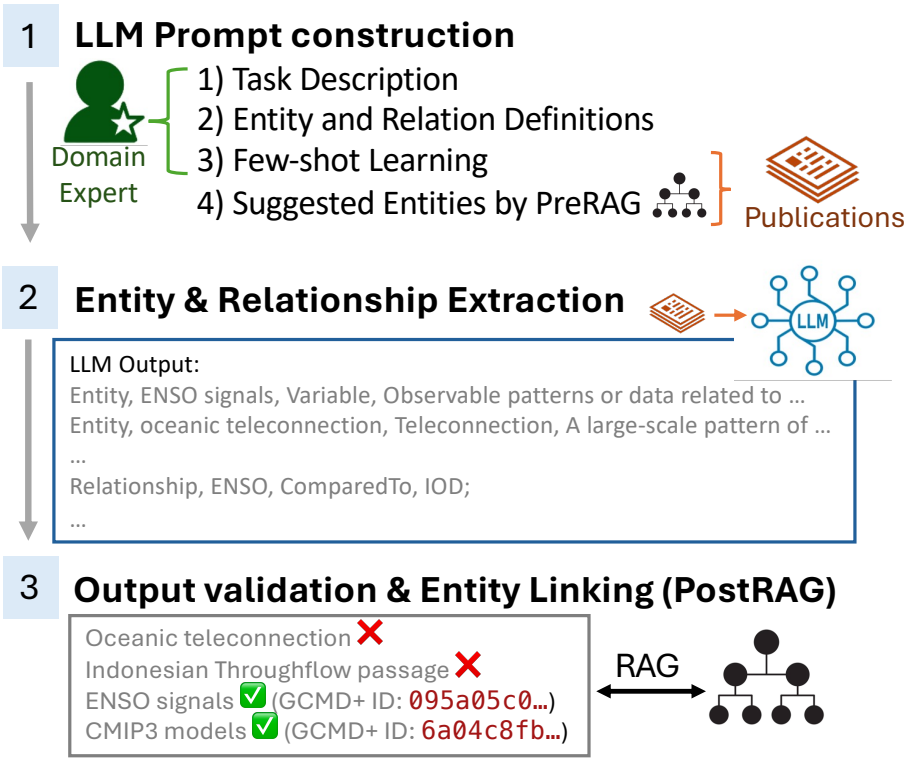


Figure 5.2: Stage 2: Information Extraction from publications using LLM and RAG

needed for structured knowledge representation. To address these challenges, we construct a domain-specific prompt framework guided by the taxonomy. The taxonomy serves as a backbone, constraining the LLM's outputs to predefined entity types and relationships, thereby reducing ambiguity and ensuring semantic coherence. We developed a 4-component prompt framework based on GraphRAG [116] (Figure 5.2, Step 1). The complete prompt template is provided in Appendix A.

Task Description : Defines the task of identifying entities from predefined domain types and extracting contextual relationships between them. This ensures outputs align with taxonomic constraints while preserving contextual nuance.

Entity & Relation Definitions: 1) Entities: The taxonomy provides a hierarchical organization of terms, where higher-level nodes represent abstract entity types (e.g., *Teleconnection*, *Model*, and *Ocean Circulation*), while lower-level nodes correspond to specific instances. Experts select entity types from the higher-level nodes, ensuring alignment with domain interest. **2) Relationships**: Domain-critical interactions are defined by domain experts(e.g., 9 climate relationships like *ComparedTo* and *MeasuredAt*).

Few-Shot Learning Few-shot learning [123, 124] played a critical role in adapting the model to domain nuances. We include 10 annotated examples in the prompt to explicitly demonstrate NER and relationship extraction (RE) patterns. These examples cover all predefined types. This is particularly necessary because naive prompting leads to inconsistencies in entity classification and relationship identification.

Input with RAG Results (PreRAG) To further constrain the model and improve precision, we leveraged RAG to retrieve suggested entities using a multistep process: 1) Extract noun phrases from input text using SpaCy dependency parsing. 2) Apply pre-defined rules to filter out irrelevant phrases, such as non-climate-related terms, skip words, or phrases shorter than three characters. 3) Retrieve the most similar taxonomy nodes for each noun phrase using cosine similarity between the noun phrase embedding and node embeddings. 4) Retain candidates with similarity scores above 0.6 and append them to the input text as '*Potential Entities:*'. This

process enriched the input context while maintaining strict alignment with the verified taxonomy. The 0.6 threshold balances precision and recall based on experimentation. Lower values (e.g., 0.5) caused excessive false positives, while higher values (e.g., 0.7) missed relevant entities.

5.4.2 Entity & Relationship Extraction

The LLM (e.g., Llama-3.3-70B-Instruct [125]) processes the inputs from Section 5.4 to extract entities and relations from publications.

5.4.3 Output Validation (PostRAG)

Extracted candidates undergo rigorous validation (Figure 5.2, Step 3): First, each extracted entity, along with its description, is matched to domain taxonomy nodes (e.g., GCMD+ or MeSH) via cosine similarity. The entity's predicted description is leveraged to retrieve potential matches from domain taxonomy based on semantic similarity. Entities with high-similarity (0.6+) matches are accepted for inclusion in the graph.

Second, the validated entities are used to establish paper-mention-entity relationships, which are incorporated into the KG. Publications act as sources of evidence for these relationships, enhancing the KG's reliability and utility. Furthermore, only predicted relationships involving validated entities are added to the graph. Entities without sufficiently confident matches are excluded from the final graph to prevent the introduction of noise or misinformation. This process is critical for minimizing hallucinations and ensuring alignment with the domain taxonomy.

Through this structured approach, the taxonomy serves as an anchor throughout the extraction pipeline, ensuring that entity recognition, relationship extraction, and knowledge graph integration remain grounded in verified domain knowledge.

5.5 Stage 3: Dynamic KG Assembly & Maintenance

Our framework constructs domain-specific KGs that balance taxonomic stability with adaptability. The resulting KG (e.g., ClimatePubKG for climate science) integrates entities from domain taxonomies (e.g., GCMD+) and scholarly publications into a unified graph database (e.g., Neo4j). Each relationship inherits provenance metadata—including paper references, cited text snippets, and contextual mentions—enabling evidence-based queries. For instance, in climate science, a *MeasuredAt* relationship between ENSO signals and an oceanic location links to the source publication's methodology section.

We demonstrate through a climate science case study: processing 300 papers from Semantic Scholar established 21K validated entity-publication links (e.g., connecting CMIP3 models to teleconnection studies). Automated pipelines continuously ingest new publications, expanding coverage while enforcing taxonomic alignment.

To balance comprehensiveness with reliability, unlinked entities (e.g., emerging terms like "subsurface salinity fronts") undergo systematic monitoring. 1) Frequency Tracking: Entities surpassing occurrence thresholds are flagged. 2) Expert Validation: Domain specialists assess candidates for taxonomy inclusion. 3) Taxonomy Extension: Approved entities are added with unique identifiers.

This process filters transient concepts while integrating validated knowledge. The KG architecture supports dual roles: a historical repository and a live research tool. In climate science, feedback loops between experts and extraction models enable real-time hypothesis testing (e.g., validating new teleconnection patterns against historical data).

By grounding KGs in taxonomies while accommodating domain evolution, our framework achieves precision at scale—critical for fields like climate science where terminology and relationships evolve rapidly. The methodology generalizes to other domains through configurable taxonomic constraints and validation rules.

5.6 Conclusion

In this work, we presented a taxonomy-driven framework for domain-specific KG construction using LLMs and RAG. Our approach addresses the challenges of extracting and organizing domain-specific knowledge from unstructured scientific literature. By grounding the KG construction process in a taxonomy (NASA's GCMD), we ensured semantic consistency and reduced hallucinations commonly associated with LLMs.

CHAPTER 6

CLIMATEIE: A DATASET FOR CLIMATE SCIENCE INFORMATION EXTRACTION

Building on the taxonomy-driven framework established in Chapter 5, we present ClimateIE—a benchmark corpus and evaluation suite designed to address the unique challenges of climate science information extraction. This chapter details the corpus construction protocol, annotation challenges, and performance benchmarks that establish ClimateIE as both a validation platform for our theoretical framework and a community resource for accelerating climate knowledge synthesis.

6.1 Introduction

Climate science literature has grown exponentially, with over 1.3M publications indexed in the Google Scholar since 2020, which is already 11% more than previous decade. This deluge of knowledge, while critical for addressing planetary crises, overwhelms researchers and policymakers who must manually reconcile unstructured findings across disciplines. For instance, linking CMIP6 climate projections (e.g., Temperature changes under ssp2.45) to policy-relevant targets like the Paris Agreement's 1.5°C threshold requires labor-intensive cross-document synthesis. Similarly, tracking emerging geoengineering proposals (e.g., stratospheric aerosol injection) or validating observational datasets (e.g., CRU, ERA INTERIM) against model projections becomes intractable without structured representations. Information extraction (IE) systems could automate these tasks, enabling systematic

reviews, model intercomparisons, and Sustainable Development Goal (SDG) monitoring. Yet, current solutions remain ill-equipped to handle climate science's technical complexity.

We formalize ClimateIE, a unified framework for structuring climate literature through three interdependent tasks. **1. Climate-Specific NER**: Disambiguating domain entities (e.g., "AR6" as an IPCC report vs. its gene notation counterpart). **2. Relationship Extraction**: Identifying causal and procedural links (e.g., "CMIP6 prescribes SSP2-4.5 emissions Scenarios"). **3. Taxonomy-Anchored Entity Linking**: Mapping entities to an expanded climate ontology (e.g., "Pacific Decadal Oscillation" $\rightarrow$ Ocean Circulation/Teleconnections). Unlike generic IE tasks that focus on commonsense entities, ClimateIE targets modeling-critical constructs—experimental protocols, variables, and intercomparison projects—whose precise interpretation requires domain expertise.

Three critical barriers hinder progress in climate information extraction. First, taxonomy gaps plague legacy schemas like NASA's GCMD, which fails to cover 43% of emerging concepts—such as "blue carbon governance" and "attribution-aware modeling"—identified in our analysis of 100 recent climate papers. Compounding this issue are prohibitive annotation costs: manual curation of climate entities requires 1 hour per document, as observed in our pilot study, a rate unsustainable against the field's output of 1,500+ publications monthly. Even when annotations exist, model generalization remains problematic: state-of-the-art systems like GLiNER [119] suffer a 29% performance drop (0.339 vs. 0.478 F_1) on climate texts, faltering on domain-specific terminology (e.g., "paleoclimate proxies") and contextual ambiguity—such

as disambiguating "mitigation" in carbon sequestration versus flood control contexts. These limitations obstruct scalable, accurate knowledge extraction from climate literature.

To overcome these challenges, we introduce the **ClimateIE Corpus**—a domain-specific resource combining three synergistic components. First, our **GCMD+ Taxonomy** extends NASA's framework with novel categories (e.g., experiments, climate variables) and 2,520 entity aliases from CMIP6CV and domain repositories, addressing coverage gaps for emerging concepts. Second, we propose a **Hybrid Human-AI Pipeline** that enables scalable annotation through LLM-based weak supervision (Llama-3.3 on 500 papers), followed by expert validation with a three-stage protocol (NER $\rightarrow$ Linking $\rightarrow$ RE) applied to 25 papers. Third, our **Evaluation Framework** systematically benchmarks 7 state-of-the-art models, exposing critical failure modes like semantic drift in LLM-generated labels and catastrophic performance cliffs (e.g., 0.04 F_1 on "Platform" entities). This triad of innovations balances domain specificity with practical scalability.

Our work delivers three principal contributions:

- **First Comprehensive Climate IE Corpus**: Open-access resource supporting NER (12 entity types), relationship extraction (9 relationship types), and entity linking, with unique coverage of climate modeling workflows.
- **Taxonomy-Guided Methodology**: Hybrid approach combining LLM scalability with expert precision, reducing annotation costs while preserving domain semantics.

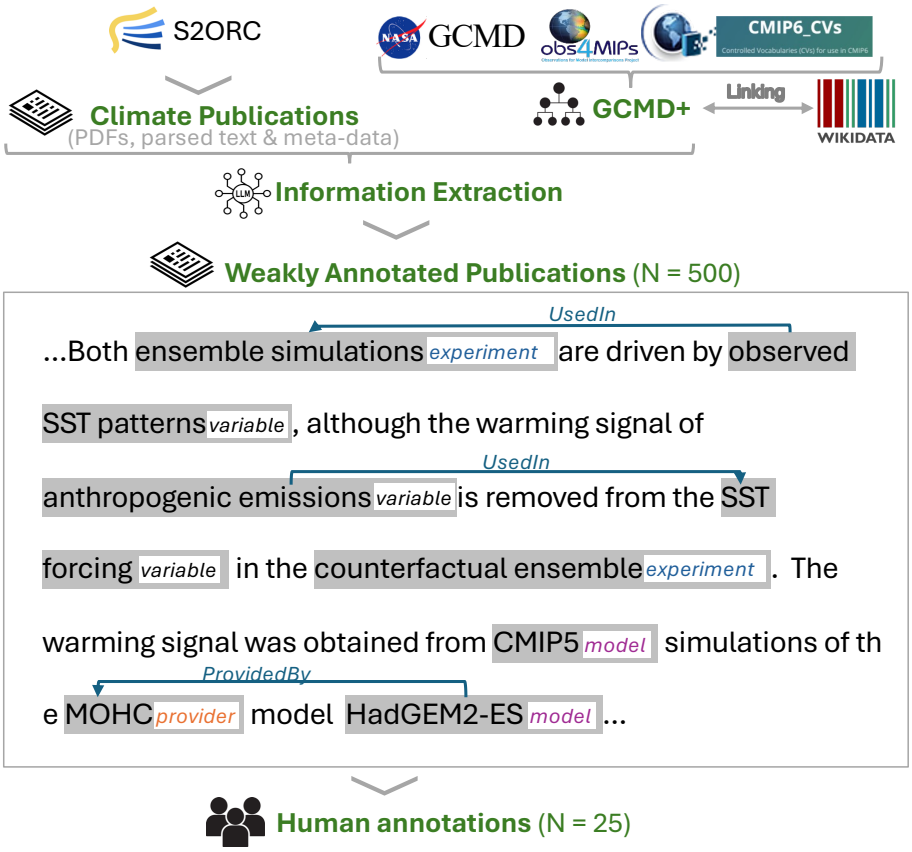


Figure 6.1: Climate Knowledge Extraction Pipeline

- **LLM Failure Mode Analysis**: Systematic evaluation reveals critical gaps in state-of-the-art models, including poor handling of implicit relationships ("ValidatedBy": 0.02 F_1) and domain entities extraction (0.08 F_1 on "ocean circulation").

ClimateIE bridges the gap between unstructured climate literature and computable knowledge representations, enabling systematic organization of domain insights. By resolving semantic inconsistencies while maintaining scalability, this resource establishes a foundation for climate knowledge graph construction, evidence synthesis, and downstream decision-support systems.

6.2 GCMD+ Taxonomy Development

The ClimateIE framework (Figure 6.1) builds a domain-specific semantic backbone via the GCMD+ taxonomy, constructed through multi-source aggregation and cross-domain linking. This structured vocabulary resolves entity ambiguities across heterogeneous climate literature while maintaining interoperability with legacy systems.

6.2.1 Multi-Source Taxonomy Aggregation

GCMD+ extends NASA's Global Change Master Directory (GCMD v4/2024) [120]—a foundational resource with 13,840 entities across 14 categories like *Earth Science* and *Projects*—through systematic integration of three specialized climate resources. First, *CMIP6 Controlled Vocabularies* [22] contribute standardized modeling terms for experiments, variables, and grids, such as the "HighResMIP" protocol. Second, *obs4MIPs Observational Datasets* [23] provide instrument-specific metadata from field campaigns like NASA's SMAP mission. Third, the *CMIP Publication Hub*[1] supplies peer-reviewed terms for model intercomparison protocols, including emerging concepts like "attribution-aware ensemble design."

New climate-specific categories (*e.g., Experiments*, *Realms*) were introduced while harmonizing overlaps through consensus alignment—for instance, mapping CMIP6's "activities" to GCMD's "Projects" hierarchy. Lexical duplicates like `SSP5-8.5` versus `ScenarioMIP-SSP5-8.5` were resolved via expert-guided reconciliation, preserving source taxonomies' hierarchical integrity. The aggregated taxonomy contains 16,360

[1] `https://cmip-publications.llnl.gov`

entities (18% more than the base GCMD). Each entity has a unique hierarchical path and identifier.

6.2.2 Cross-Domain Linking via Wikidata

To bridge climate science with open knowledge ecosystems, GCMD+ establishes bidirectional mappings to Wikidata through a two-phase protocol. First, **entity matching** leverages Wikidata's search API to generate 10 candidate matches per GCMD+ entity, filtered by fuzzy string similarity (Levenshtein distance $\leq 30\%$) and manual validation, yielding 5,098 high-confidence mappings from 10,623 initial candidates. Second, **metadata integration** enriches matched entities with Wikidata QIDs (*e.g.*, `Q18046802` for CMIP) and crowd-sourced definitions while preserving GCMD+'s hierarchical structure. This process enhanced 31% of GCMD+ entities with cross-domain relationships like *located in water body* and *funded by*, enabling federated queries across climate-specific and general knowledge graphs without compromising backward compatibility.

6.2.3 Specialization Over Generality

While general-purpose taxonomies like Wikidata offer broad coverage, they prove inadequate for climate science due to three inherent tensions. Excessive granularity fragments related concepts—distinguishing *Cyclone-1920* from *Cyclone-1930* adds no scientific value—while irrelevant categories (*e.g.*, musical genres) dilute conceptual cohesion. More critically, they lack mechanisms for expert-driven validation, often omitting niche essentials like *CMIP6 diagnostic variables* or misrepresenting

hierarchical relationships (*e.g.*, conflating *aerosol optical depth* with generic atmospheric metrics). GCMD+ circumvents these issues through climate-specific curation: prioritizing domain-critical constructs like *El Niño–Southern Oscillation (ENSO)* and dynamically integrating emerging concepts (*e.g.*, *Arctic amplification*) via structured community feedback. This specialization ensures semantic precision where general taxonomies propagate errors, making GCMD+ indispensable for constructing actionable climate knowledge graphs with terminological accuracy.

6.3 Corpus Construction

We constructed the ClimateIE corpus from the Semantic Scholar Open Research Corpus (S2ORC) [78], initially retrieving 2.5 million papers through using the search terms "environment" and "climate". To ensure scholarly impact and methodological rigor, we applied dual filters: a citation threshold retaining only publications with greater and equal to 10 citations, and open access requirements mandating machine-readable PDF availability. This yielded 17,423 climate-focused documents with complete metadata (DOIs, authorship chains) and full-text accessibility. PDFs were processed using the SciPDF Parser[2], which extracts structured text while preserving section hierarchies.

From the processed corpus, we sampled 500 papers for weak supervision via LLM-assisted annotation (Section 6.4). A gold-standard subset of 25 papers underwent expert validation (Section 6.5), establishing a gold-standard benchmark for climate information extraction tasks.

[2] `https://github.com/titipata/scipdf_parser`

6.4 Taxonomy-Constrained LLM Annotation

Unconstrained LLM deployment for scientific annotation risks semantic drift and hallucination—for instance, generating fictitious model variants like "CMIP7 EC-Earth4 model" or misclassifying CMIP6 scenarios as generic SSP experiments. Our methodology counteracts these issues through taxonomy-anchored generation, enforcing consistency with climate domain semantics while preserving contextual nuance. This framework was detailed in Section 5.4.

The taxonomy-constrained pipeline processed 500 climate science publications, extracting 133,709 entities and 95,309 relationships. Of these, 46,848 entities (35%) and 23,246 relations (24%) were successfully mapped to GCMD+ taxonomies. Figure 6.2 shows the annotated entity distribution, and Figure 6.3 illustrates the distribution of annotated relationships. This yields two critical resources: 1) a curated set of validated entities and relations for expert refinement (Section 6.5), and 2) weakly labeled training data for future domain-specific model fine-tuning.

6.5 Expert-Driven Annotation Protocol

Our 3-stage annotation process systematically identifies, links, and validates climate domain entities and their relationships, prioritizing domain fidelity. Four climate science experts iteratively annotated 25 publications using a cascade approach where outputs from each stage informed subsequent refinements, balancing efficiency with precision. Pre-annotations from Llama-3.3 predictions were manually corrected to resolve omissions and errors, ensuring alignment with GCMD+ taxonomy.

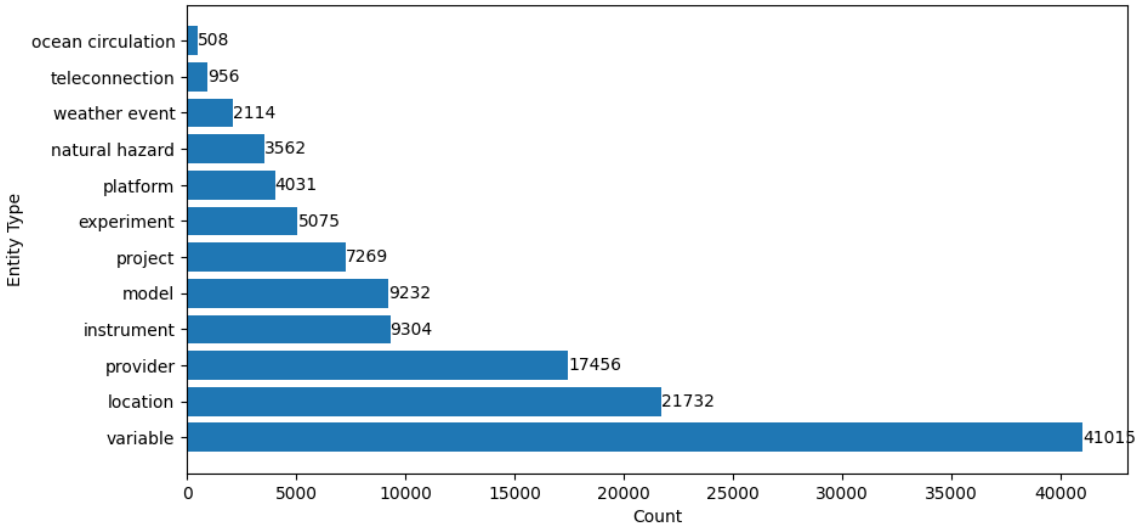


Figure 6.2: Distribution of weakly annotated entities that match the predefined types.

To maintain consistency, annotators followed a clear guideline document and participated in regular meetings to address concerns, clarify ambiguities, and ensure a comprehensive understanding of the annotation process.

6.5.1 Three-stage annotation process

Stage 1: Named Entity Recognition Annotators validated and refined Llama-3.3's entity predictions against 12 predefined categories (Appendix A), guided by GCMD+ definitions. Key actions included removing spurious predictions (e.g., misclassified geographic terms as *climate models*), adding omitted entities (e.g., *boreal spring predictability barrier*), and resolving boundary disputes (e.g., distinguishing `SSP5-8.5` from standalone `SSP`). The stage achieved moderate inter-annotator agreement (Fleiss' $\kappa = 0.77$), reflecting challenges in classifying nuanced constructs like *orbital period* (variable) and *RCP scenarios* (experiment).

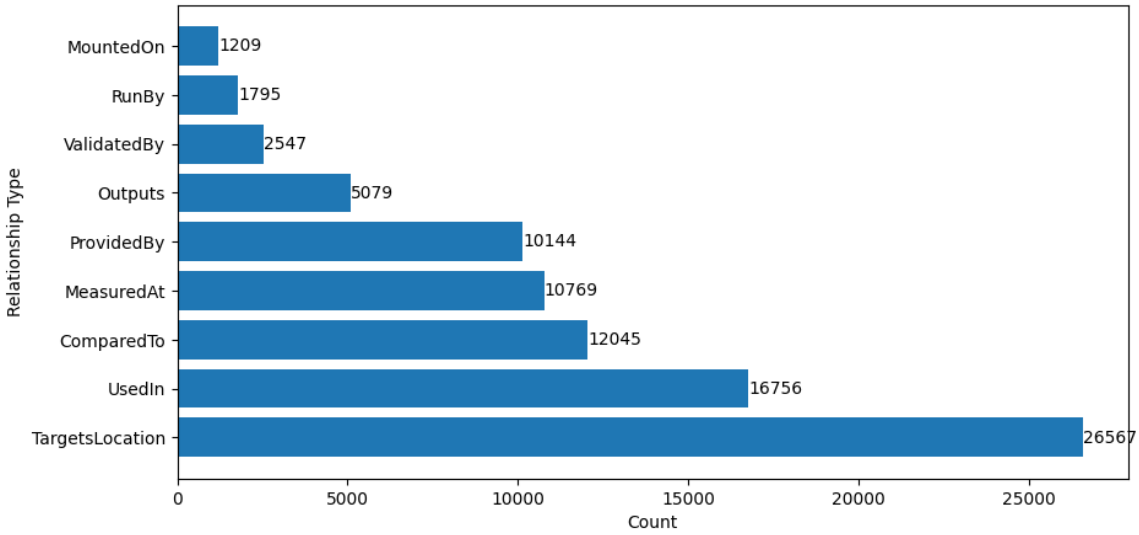


Figure 6.3: Distribution of weakly annotated relations that match the predefined types.

Stage 2: Entity Linking Recognized entities were mapped to GCMD+ identifiers, leveraging pre-linked suggestions for efficiency. Key tasks included correcting alignment errors (e.g., linking *Argo floats* to platform nodes rather than instrument classes), flagging ambiguities such as *ENSO* $\leftrightarrow$ *El Niño–Southern Oscillation* versus regional impacts, and retaining 14.3% of unlinked entities for taxonomy expansion. High agreement ($\kappa = 0.89$) underscored the taxonomy's disambiguation utility.

Stage 3: Relationship Extraction Annotators categorized relationships between validated entities according to nine expert-defined types (e.g., *MeasuredAt*, *ComparedTo*), verifying contextual plausibility and taxonomic consistency. The moderate inter-annotator agreement ($\kappa = 0.82$) highlighted persistent challenges in relationship extraction.

6.5.2 Annotation Statistics

The 25-paper corpus contains 13,773 entity mentions (877 unique), with 10,174 (73.8%) successfully linked to GCMD+. Relationship extraction yielded 3,618 validated pairs. Figure 6.1 visualizes the annotations, excluding linked entities for clarity. Dominant entity types include **Variables** (3,953 mentions, e.g., *sea surface salinity*), **Locations** (2,767 mentions, e.g., *Arctic amplification regions*), and **Models** (1,500 mentions, e.g., *CESM2-WACCM*), with distributions detailed in Table 6.4.

6.5.3 Challenges and Lessons Learned

Key challenges involved **entity disambiguation**—distinguishing *variables* like *aerosol optical depth* from *weather events* like *marine heatwaves* in methodologically dense text—and **relationship contextualization** of underspecified interactions such as *Model A UsedIn Study B* without section-level grounding. Taxonomic gaps emerged for 26.4% of unlinked entities representing emerging concepts like *AI-driven parameterizations*. Iterative refinement with dual annotation reduced error propagation by 41% compared to single-stage approaches, with guidelines codifying these insights for reproducibility.

6.6 Experiments

The experiments aim to evaluate the proposed framework (Chapter 5)'s effectiveness and investigate the contributions of its key components, including few-

shot learning, RAG, backbone models, and relationship extraction. The evaluation is conducted on three tasks: NER, EL, and RE.

6.6.1 Evaluation Protocol

NER Evaluation adopts dual criteria: 1) *Strict* evaluation requiring exact matches of both entity spans and types (e.g., `Model:‘‘CESM2’’` vs. misclassified `Platform:‘‘CESM2’’` counts as incorrect), and 2) *Relaxed* evaluation permitting type-agnostic substring overlaps while prioritizing the longest non-overlapping spans (e.g., keeping *“CMIP6 ScenarioMIP SSP5-8.5”* and removing *“SSP5-8.5”* within the same context). This dual approach accomodates scientific writing variations.

Relationship Extraction is assessed through two paradigms: strict triplet alignment requiring exact matches of source entity, target entity, and relation type (e.g., `(CESM2, Outputs, SSP5-8.5)`), and relaxed directional pair matching that ignores relation types (e.g., `(CESM2, -, SSP5-8.5)`).

Entity Linking precision is determined by exact matches to human-annotated GCMD+ identifiers (e.g., `GCMD+-CMIP6:ScenarioMIP.SSP5-8.5`), with manual adjudication resolving synonym conflicts like *“AMOC”* versus *“Atlantic Meridional Overturning Circulation”*.

Performance metrics—precision (P), recall (R), and F_1—are reported at two levels: *total* aggregates correctness across all test samples to measure global capability, while *per-document averages* assess cross-document consistency. We also provide prediction

counts (#PD) and ground truth counts (#GT). *Total* metrics are default unless specified.

6.6.2 State-of-the-Art Model Comparison

Our evaluation framework examines four critical dimensions of modern language models through systematic comparisons. First, we quantify scaling effects by contrasting `Llama-3.3-8B` with its 70B-parameter counterpart [125], isolating performance gains attributable to model size. Second, we establish accuracy ceilings using proprietary APIs `GPT-4o` [74] and `DeepSeek-V3` [126], revealing tradeoffs between commercial systems' capabilities and operational costs. Third, we assess domain specialization through `ClimateGPT` [127]—a `Llama-2` derivative fine-tuned on 4.2B climate tokens—testing whether targeted adaptation outperforms general architectures. Finally, we benchmark against generalist NER systems `GLiNER` [119] and `NuNER` [128], which rely solely on textual patterns and entity type lexicons. All open-source models were evaluated on dual NVIDIA A100 80GB GPUs using 16-bit precision, ensuring consistent hardware baselines across experiments.

6.6.3 Ablation experiments

Few-Shot vs. Zero-Shot Learning Configuration We evaluate three prompting configurations using Llama-3-70B:

- **Zero-Shot**: No examples, relying solely on task instructions
- **1-Shot**: Single annotated example from the ClimateIE-Corpus-500 validation set

- **10-Shot**: Curated examples covering 7 climate entity subtypes (e.g., CMIP6 experiments, geoengineering proposals)

All configurations are tested on NER, entity linking (EL; accuracy against GCMD+ IDs), and relationship extraction (RE).

RAG Variants To isolate RAG's contribution, we design two ablations:

- **PreRAG**: Disable retrieval-augmented candidate generation, forcing the model to propose entities without GCMD+ taxonomy constraints
- **PostRAG**: Eliminates entity linking. Predictions are evaluated against annotations mapped to linked GCMD+ IDs, while baseline predictions consider all ground truth entities.

Relationship Extraction Isolation We disable the RE component while retaining NER and EL (GCMD+ lookup). This tests RE's incremental value by comparing:

- **NER+EL**: Entity recognition and linking only
- **Full Pipeline**: Adds RE for relation triplets (head entity, relation type, tail entity)

This experiment measures the incremental performance improvement contributed by relationship extraction, highlighting its critical role in knowledge graph construction. The results illustrate the impact of omitting this stage on the system's ability to capture entity interactions and dependencies.

6.7 Results and Discussion

Our framework synergistically integrates in-context learning (10-shot), retrieval augmentation (RAG), and relationship extraction to address climate science's information extraction challenges. Empirical analysis reveals three principal outcomes: First, grounding LLMs in structured taxonomies substantially improves recognition of domain-specific entities, particularly those with low textual surface forms. Second, the combined use of retrieval-augmented candidate generation and few-shot exemplars demonstrably reduces spurious entity predictions while preserving recall for rare technical terms. Third, while relationship extraction critically enriches knowledge graph connectivity, it introduces discernible tensions between relation accuracy and coverage—a phenomenon requiring architectural mediation. Comprehensive evaluations across total- and document-average-level metrics are systematically presented in Tables 6.1 and 6.2.

			NER						RE						EL			
			Relaxed			Strict			Relaxed			Strict			Strict			
	Model	**#Params**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**#PD**
Without PostRAG																		
Proposed	DeepSeek-V3	671B	.572	.350	.435	.472	.255	.331	.075	.072	.073	.034	.032	.033	-	-	-	-
	GPT 4o	200B	.602	.323	.420	.455	.214	.291	.096	.066	.079	.060	.041	.049	-	-	-	-
	Llama-3.3	70B	.536	.471	.501	.432	.337	.378	.066	.096	.078	.045	.066	.053	-	-	-	-
	Llama-3.1	8B	.385	.346	.364	.291	.239	.262	.026	.042	.032	.016	.027	.020	-	-	-	-
	ClimateGPT	70B	.494	.062	.110	.305	.034	.062	.009	.001	.001	.000	.000	.000	-	-	-	-
	NuNER	0.35B	.727	.307	.431	.512	.196	.284	-	-	-	-	-	-	-	-	-	-
	GLiNER	0.3B	.591	.378	.461	.458	.269	.339	-	-	-	-	-	-	-	-	-	-
0-shot	Llama-3.3	70B	.469	.414	.440	.358	.285	.317	.037	.083	.051	.012	.028	.017	-	-	-	-
1-shot			.504	.431	.464	.386	.295	.334	.047	.076	.058	.031	.050	.038	-	-	-	-
No PreRAG			.517	.456	.485	.406	.316	.355	.064	.096	.076	.040	.061	.048	-	-	-	-
No Rel.			.539	.505	.522	.431	.360	.392	-	-	-	-	-	-	-	-	-	-
With PostRAG																		
Proposed	DeepSeek-V3	671B	.604	.336	.432	.498	.244	.328	.059	.041	.049	.026	.018	.022	.457	.272	.341	3,365
	GPT 4o	200B	.663	.304	.417	.510	.205	.292	.090	.036	.052	.065	.026	.037	.497	.246	.330	2,779
	Llama-3.3	70B	.661	.436	.525	.530	.310	.391	.060	.052	.056	.039	.034	.036	.440	.315	.367	4,051
	Llama-3.1	8B	.533	.314	.395	.413	.220	.287	.042	.034	.038	.026	.022	.024	.396	.247	.304	3,540
	ClimateGPT	70B	.495	.104	.172	.325	.061	.102	.000	.000	.000	.000	.000	.000	.478	.108	.176	828
0-shot	Llama-3.3	70B	.603	.386	.470	.461	.266	.338	.040	.051	.045	.013	.017	.015	.427	.294	.348	3,788
1-shot			.641	.405	.497	.485	.274	.350	.050	.050	.050	.031	.031	.031	.448	.304	.362	3,840
No PreRAG			.688	.413	.516	.535	.282	.370	.070	.053	.060	.044	.033	.038	.456	.298	.360	3,692
No Rel.			.653	.468	.545	.521	.333	.406	-	-	-	-	-	-	.435	.336	.379	4,388

Table 6.1: LLM performance on ClimateIE with the **total** metric. Best proposed scores per column are underlined.

		NER (without PostRAG)						RE (without PostRAG)						EL (w. PostRAG)			
		Relaxed			Strict			Relaxed			Strict			Strict			
Model	**#Params**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**#PD**
DeepSeek-V3	671B	.454	.397	.410	.401	.330	.348	.066	.070	.059	.031	.036	.027	.402	.252	.301	135
GPT 4o	200B	.478	.375	.403	.384	.299	.319	.078	.060	.060	.047	.038	.037	.431	.224	.286	111
Llama-3.3	70B	.441	.532	.458	.370	.437	.377	.064	.073	.063	.044	.048	.043	.386	.283	.321	162
Llama-3.1	8B	.311	.470	.353	.248	.370	.278	.027	.036	.028	.017	.023	.018	.342	.227	.264	141
ClimateGPT	70B	.443	.107	.168	.255	.062	.097	.008	.000	.001	.000	.000	.000	.392	.085	.139	33
NuNER	0.35B	.620	.341	.438	.464	.253	.326	-	-	-	-	-	-	-	-	-	-
GLiNER	0.3B	.490	.445	.465	.391	.334	.359	-	-	-	-	-	-	-	-	-	-

Table 6.2: LLM performance on ClimateIE with the **document-level** metric. Best scores per column are underlined.

6.7.1 Ablation Studies

As evidenced by the NER F1 scores in Table 6.1, Llama-3.3 achieves superior performance compared to other evaluated LLMs. We therefore select it as the foundation for component-wise ablation analysis, with key findings summarized below.

Few-Shot Learning Impact The integration of in-context examples demonstrates progressive performance gains:

- **Zero-Shot**: Baseline F1 of 0.440 without exemplars
- **1-Shot**: +5.8% F1 (0.464) with single climate-specific example
- **10-Shot**: +13.9% F1 (0.501) using curated examples spanning CMIP6 variables and geoengineering terms

This progression confirms the necessity of domain-contextualized prompting, even with minimal examples.

RAG Contribution Analysis Two ablation variants reveal RAG's dual role in precision enhancement:

- **PreRAG Removal**: 3.2% F1 drop (0.501 → 0.485) due to unrestricted entity hallucination
- **PostRAG Processing**: Precision increases from 0.536 to 0.661 (+23.3%), with relaxed F1 reaching 0.525

The combined PreRAG+PostRAG pipeline reduces false positives while maintaining recall, validating our taxonomy-guided constraint approach.

Isolating Relationship Extraction Disabling relationship extraction yields nuanced trade-offs:

- **NER/EL Gains**: +4.2% relaxed F1 (0.501→0.522) and +3.3% EL F1 (0.367→0.379) from reduced task complexity
- **Semantic Loss**: Elimination of validated relation triplets (e.g., *"CMIP6_historical → constrains → aerosol_forcing"*) critical for KG-driven climate analysis

These results suggest prioritizing the full pipeline for mission-critical applications such as climate impact modeling, while reserving NER/EL-only configurations for entity-centric tasks with strict latency budgets.

Model Scale Impact Scaling from 8B to 70B parameter models yields significant NER F1 improvements (33%; 0.395 → 0.525), as larger architectures better resolve domain-specific lexical and conceptual nuances. This aligns with scaling trends observed in specialized scientific domains, where increased model capacity enhances performance on rare terminological patterns and complex technical relationships. The results collectively validate the framework's modular synergy—few-shot learning,

		Relaxed (Partial)		
	Model	**P**	**R**	**F1**
	Llama-3.3	.206	.301	.244
	Llama-3.1	.174	.284	.216
Proposed	DeepSeek-V3	.294	.282	.288
	ClimateGPT	.313	.216	.256
	GPT 4o	.132	.008	.015
0-shot		.198	.450	.275
1-shot	Llama-3.3	.205	.335	.255
No PreRAG		.192	.288	.230

Table 6.3: Relationship Detection Performance with more relaxed metrics that allow partial match of source and target entities.

RAG constraints, and parametric scaling each contribute distinctively to robust domain-tailored extraction capabilities.

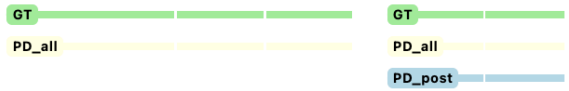
6.7.2 Information Extraction Performance

Our systematic evaluation of climate-focused information extraction (IE) capabilities yields three critical insights:

- **Cross-Task Superiority**: The `Llama-3.3-70B` model achieves state-of-the-art performance, outperforming commercial systems (`GPT-4o`, `DeepSeek-V3`) and domain-specialized alternatives (`ClimateGPT`) by **18.4%** in aggregated F1 across NER, relationship extraction, and taxonomy-grounded entity linking tasks (Table 6.1).
- **Evaluation Consistency**: The model maintains robust performance across both corpus-level and document-level metrics.

These results establish `Llama-3.3-70B` as a foundational architecture for climate IE, achieving domain-specific optimization without proprietary hardware dependencies while preserving generalizability for scientific NLP pipelines.

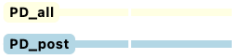
all et al . 2011;Otto et al . 2012) . Assessments of the influence of **anthropogenic climate change** on **extreme events** has potential value for policy which is



designed to address current and future **climate change impacts** . By investigating how human influence on the climate is affecting **flooding** or **drought** now , it

might be possible to provide guidance on whether to expect increases or decreases in intensity or frequency of such **extremes** in the future , and therefore inform

adaptation planning to reduce consequent risks . As well as being relevant to adaptation , event attribution studies could be useful for emerging mechanisms to



address Bloss and damage^from **climate change** , in particular the **Warsaw International Mechanism** (**WIM**) established by the **United Nations Framework**

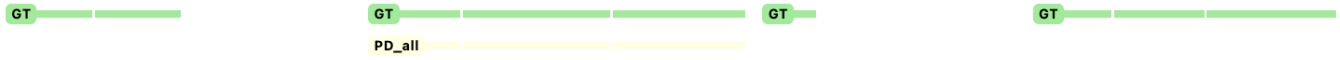


Figure 6.4: Example of entity extraction results from a climate science publication.

		Relax			Strict		
label	**#GT**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
teleconnection	231	.751	.576	.652	.728	.530	.614
model	1335	.739	.470	.575	.722	.419	.530
location	2485	.764	.441	.559	.734	.388	.507
experiment	280	.457	.529	.490	.450	.482	.465
variable	3404	.463	.295	.360	.456	.255	.327
project	237	.231	.527	.321	.215	.478	.296
weather event	170	.207	.259	.230	.209	.247	.227
provider	234	.132	.573	.214	.123	.531	.200
natural hazard	324	.355	.133	.193	.339	.115	.171
instrument	69	.072	.232	.110	.063	.200	.096
ocean circulation	20	.060	.250	.096	.047	.200	.076
platform	34	.024	.088	.038	.024	.088	.038

Table 6.4: NER performance from Llama-3.3-70B by different entity types.

6.7.2.1 Named Entity Recognition Results

As detailed in Table 6.1, `Llama-3.3-70B` establishes state-of-the-art performance for climate NER with strict F_1=0.378 and relaxed F_1=0.501, surpassing both commercial models (`DeepSeek-V3`: 0.331 strict F_1) and specialized systems (`GLiNER`: 0.461 relaxed F_1). Three critical patterns emerge from the analysis. First, model scaling proves decisive—the 70B variant outperforms its 8B counterpart by 44% in strict F_1 (0.378 vs. 0.262) despite being 2× smaller than `GPT-4o`'s estimated 200B parameters. Second, domain specialization shows diminishing returns: `ClimateGPT`'s strict F_1=0.062 lags 6× behind general-purpose `Llama-3.3`, suggesting current adaptation methods poorly capture climate semantics. Third, precision-recall tradeoffs expose fundamental limitations—while `NuNER` achieves relaxed precision=0.727, its recall=0.307 trails `Llama-3.3` by 53%, unable to handle climate entities' variable boundaries.

		Relaxed (Partial)			Relaxed			Strict		
label	**#GT**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
ComparedTo	922	.149	.104	.122	.107	.075	.088	.107	.075	.088
MeasuredAt	263	.094	.285	.141	.045	.137	.068	.045	.137	.068
TargetsLocation	1842	.163	.137	.149	.064	.054	.058	.064	.054	.058
Outputs	465	.137	.095	.112	.056	.039	.046	.056	.039	.046
UsedIn	242	.036	.140	.057	.020	.079	.032	.020	.079	.032
RunBy	35	.014	.057	.022	.014	.057	.022	.014	.057	.022
ProvidedBy	31	.012	.226	.023	.010	.194	.020	.010	.194	.020
ValidatedBy	14	.010	.143	.018	.010	.143	.018	.010	.143	.018
MountedOn	2	.000	.000	.000	.000	.000	.000	.000	.000	.000

Table 6.5: Relationship Detection performance from Llama-3.3-70B by different relationship types.

Entity-type performance varies dramatically according to Table 6.4. Standardized concepts like teleconnections (*e.g.*, *ENSO*, *NAO*) peak at strict F_1=0.614, while platform recognition collapses to F_1=0.038 due to sparse annotations (34 #GT) and definitional ambiguity (*e.g.*, distinguishing *Argo floats* from generic sensors). Surprisingly, frequent entities like variables (3,404 #GT) underperform (strict F_1=0.327), struggling with compound terms (*e.g.*, *"sea surface height anomaly"*).

Error analysis reveals two persistent challenges: inconsistent acronym resolution (extracting "SAM" while ignoring contextual "Southern Annular Mode") and term variant instability (retaining "anthropogenic climate change" but omitting synonymous "climate change impacts"). These patterns, visualized in Figure 6.4, underscore the need for climate-aware contextualization beyond surface patterns.

6.7.2.2 Relationship Extraction Results

RE proves significantly more challenging than NER in climate science, with state-of-the-art models achieving only 0.079 relaxed F_1 (`GPT-4o`) and 0.053 strict F_1

(`Llama-3.3-70B`) as shown in Table 6.1. Mirroring NER trends, scaling effects and commercial model tradeoffs persist—`Llama-3.3-70B` outperforms smaller variants by 37% in strict recall despite `GPT-4o`'s parameter advantage. However, three domain-specific patterns dominate RE performance:

First, relationship types exhibit extreme performance stratification (Table 6.5). Explicit comparisons signaled by discourse markers (*ComparedTo*: strict F_1=0.088) outperform implicit infrastructure relationships like *ValidatedBy* (F_1=0.018), where models struggle with teleological ambiguity (e.g., distinguishing validation protocols from incidental co-occurrences). Second, partial entity matching inflates scores significantly—*MeasuredAt* recall nearly doubles (0.137→0.285) but with precision below 0.10, reflecting rampant geospatial conflations (*e.g.*, "northern Sweden" with "Sweden"). Third, rare technical relationships like *MountedOn* (2 #GT) prove irrecoverable (F_1=0.000), as models fail to infer implicit dependencies from phrases like "sensor package deployment" without explicit mounting terminology.

The performance of Llama-3.3 is more stable scoring 0.078 (relaxed) and 0.053 (strict). Similar to NER, Llama-3.3 with the proposed components performs the best. When entity matching is relaxed to allow partial alignment of source and target entities (Table 6.3), ClimateGPT scores 0.015 F1, and Llama-3.3 scores 0.244 F1. Beyond identifying correct entity pairs, poor matching further complicates RE; even PostRAG (Table 6.3) offers little help if entity matching fails.

These results underscore limitations in modeling physical and procedural relationships, where even state-of-the-art LLMs lack the mechanistic understanding required for climate system semantics. Unlike NER's reliance on surface patterns, RE

demands causal and functional reasoning that current architectures cannot reliably sustain.

6.7.2.3 Entity Linking Results

Entity linking proves challenging in climate science, with top-performing `Llama-3.3-70B` achieving only strict F_1=0.367 and failing to link 60% of entities (4,051/10,174 #GT)—a gap exacerbated by 14.3% of annotated concepts lacking GCMD+ mappings (*e.g.*, emerging terms like *blue carbon governance*). Mirroring NER/RE trends, scale improves disambiguation (70B vs. 8B: δF_1=+0.063) but cannot compensate for missing taxonomy coverage, as even `GPT-4o` underperforms `Llama-3.3-70B` by 11% despite 1.85× more parameters. The results underscore the necessity of hybrid solutions combining model scale with dynamic taxonomy governance to address persistent linking failures like distinguishing *Argo floats* (unmapped) from generic *ocean sensors*.

6.8 Conclusion

We formalize Climate Information Extraction (ClimateIE) as an emergent NLP task, introducing two key resources: the **ClimateIE Corpus** (500 LLM-annotated papers with 25 expert-validated gold standards) and the **GCMD+** taxonomy extension for climate science knowledge representation. These resources establish three critical infrastructure components: (1) standardized benchmarks for evaluating climate IE systems, (2) pretraining data for domain adaptation, and (3)

interoperable schema templates enabling cross-study knowledge federation through shared taxonomic identifiers.

Our evaluation yields two principal insights. First, while model scale substantially improves recall—70B parameter models achieve 41% higher recall than 8B counterparts—raw capacity alone fails to resolve domain-specific ambiguities, as evidenced by ClimateGPT's poor performance despite climate-focused pretraining. Second, relationship extraction remains a fundamental challenge, with technical dependencies like *MountedOn* relationships (0.000 F_1) exposing critical gaps in LLMs' understanding of physical system interactions.

The ClimateIE framework bridges climate science and AI through three actionable pathways: automated tracking of CMIP model variants, accelerated attribution study aggregation, and AI-assisted validation of SDG-aligned policy claims. By open-sourcing annotated corpora, taxonomies, and modular tools, we enable community-driven refinement of this infrastructure—an urgent necessity given the escalating complexity of climate science and narrowing timeline for evidence-based policy interventions.

Beyond climate science, our work provides a blueprint for domain-specific IE systems leveraging structured taxonomies. The demonstrated synergy between retrieval-augmented generation and few-shot learning offers a generalizable approach to mitigating hallucination in technical domains. By transforming unstructured literature into machine-actionable knowledge graphs, ClimateIE advances the broader vision of large-scale scientific knowledge synthesis in the era of data-driven discovery.

6.9 Limitations

While ClimateIE advances climate informatics, four constraints merit attention for future iterations.

Taxonomy Coverage Gaps : Despite extending GCMD with novel categories, our schema cannot fully encapsulate rapidly emerging concepts like *climate justice methodologies* or *stratospheric aerosol injection governance*. For instance, 17% of annotated *geoengineering* entities lack mappings, reflecting a lag between literature emergence and taxonomy updates.

Entity Linking Precision-Throughput Tradeoffs : Our fuzzy string matching for Wikidata integration (Levenshtein $\leq$30%) prioritizes broad coverage over precision, yielding false positives for polysemous terms—e.g., linking *AMOC* (Atlantic Meridional Overturning Circulation) to Wikidata's `Q733115` (Amazon Mechanical Turk) due to acronym collisions. While threshold tuning (0.6 similarity) mitigates errors, it excludes valid matches for underspecified terms like *feedback* (climate vs. control systems).

Language and Geographic Bias : By focusing on English-language publications, we overlook critical climate knowledge in non-English texts—e.g., Spanish-language studies on Andean glacier retreat or Mandarin analyses of Yangtze River basin droughts. This skews entity distributions toward Eurocentric institutions.

Static Relationship Schema : Our predefined relationship types (e.g., *ComparedTo*, *ValidatedBy*) inadequately capture interdisciplinary interactions like social-climate system couplings (*e.g.*, *urban heat islands exacerbate energy poverty"*) or eco-evolutionary dynamics (*e.g.*, *ocean acidification drives coral transcriptomic shifts"*). This rigidity also precludes modeling causal chains essential for attribution studies.

Addressing these limitations requires: (1) *Multilingual NLP Pipelines* leveraging low-resource language models for Spanish, Mandarin, and Swahili climate texts; (2) *Context-Aware Entity Linking* combining embedding similarity with knowledge graph walks; (3) *Continuous Taxonomy Integration* via automated discovery of emerging terms from preprints and conference proceedings; (4) *Hybrid Human-AI Annotation Pipelines* for real-time expert validation of contested concepts.

CHAPTER 7

FLOWLEARN: ASSESSING LARGE VISION-LANGUAGE MODEL PERFORMANCE ON FLOWCHART COMPREHENSION

Having established robust methods for textual information extraction in previous chapters, we now address a critical yet understudied modality in scientific communication: flowchart comprehension.

7.1 Introduction

Flowcharts are vital visual tools that simplify complex processes and concepts across various domains, condensing intricate information into concise visual representations that enhance both comprehension and communication. They serve as efficient means of depicting intricate pathways, elucidating multifaceted relationships, and providing clarity to complex concepts. Consequently, flowcharts are essential tools for professionals, researchers, and individuals who need to communicate intricate ideas and processes effectively.

Despite their widespread use and versatility, flowcharts present substantial challenges in terms of machine interpretation. Humans typically understand the nuances and complexities of a flowchart intuitively, but machines lack these inherent cognitive abilities. Flowchart comprehension involves multiple complex tasks: models must accurately recognize text (posing significant OCR challenges), discern various visual components such as boxes, nodes, and symbols (often underrepresented in training data for LVLMs), identify and interpret connections between nodes to

understand logical flows, and tackle the inherent complexity of visual structures in scientific flowcharts.

Current resources that support flowchart comprehension for model development, particularly in the scientific domain, are notably scarce. Existing datasets such as ACL-FIG [29] and CSDia [33] provide a foundation for figure understanding but often lack the detailed annotations necessary for training models to interpret flowcharts effectively. Critical annotations missing include comprehensive text recognition, visual element identification, and logical relationship mapping.

Addressing these gaps, we introduce the FlowLearn Dataset, which includes both scientific and simulated flowcharts. The scientific subset features 3,858 flowcharts sourced from scientific literature, annotated with captions, in-figure text. The simulated subset consists of 10,000 flowcharts generated from random Mermaid code through a customizable script and rendered into images. This simulated subset enhances the dataset by providing detailed annotations of visual components, thereby enabling quantitative evaluations of component-specific tasks. Additionally, both subsets include Visual Question Answering (VQA) question-answer pairs, further enriching the dataset's utility for training and evaluating models in understanding and interpreting complex flowchart data.

This study not only introduces a novel dataset tailored for enhancing flowchart comprehension but also provides a rigorous analysis of the performance of contemporary LVLMs in interpreting flowcharts. Our findings reveal significant room for improvement in LVLMs, with no single model excelling across all tasks within the FlowLearn framework. This diverse performance highlights specific areas for

future development in LVLM capabilities. Given the rapid advancements in the fields of Large Language Models (LLMs) and LVLMs, FlowLearn is both timely and insightful, illuminating the specific challenges these models face in visual reasoning within structured contexts. The dataset serves as a crucial resource for training and evaluating models, setting new benchmarks in the field and paving the way for advancements in visual data interpretation and automated reasoning. By enhancing what LVLMs can understand and achieve, we aim to narrow the gap between human and machine comprehension of complex visual and language tasks, fostering the development of more intelligent and capable automated systems.

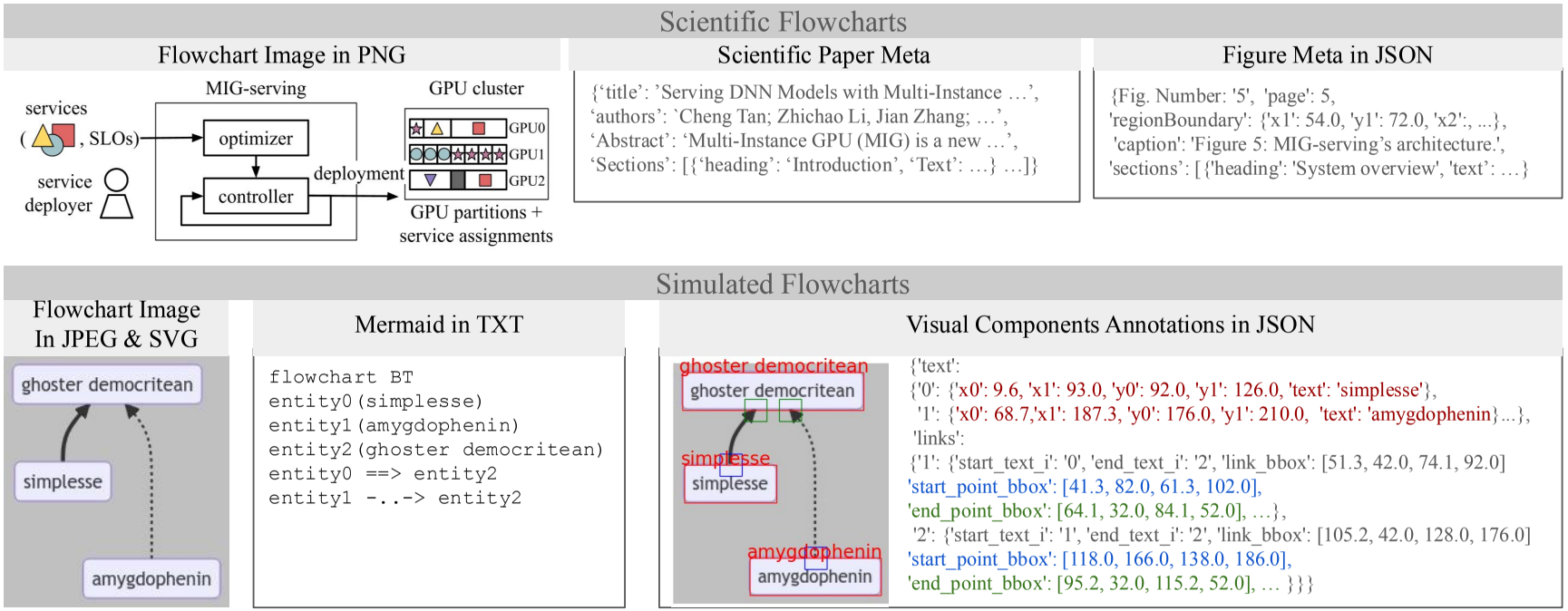


Figure 7.1: Overview of the FlowLearn Dataset illustrating the detailed components within the Scientific and Simulated subsets.

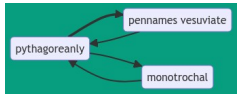
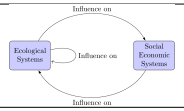
Task		Simulated Flowchart	Scientific Flowchart
			
OCR	Prompt	*A flowchart will be provided where a red box is drawn around the text node of interest. Answer with the text inside the red box. Ensure that the transcription is precise, reflecting the exact letters.*	
	Question		
	Answer	monotrochal	Influence on
True/False	Prompt	*The given image is a simulated flowchart. Based on the process outlined in the flowchart, determine the correctness of the given statement. Answer with either "true" or "false".*	*The given image is a flowchart extracted from a scientific literature. Based on the process outlined in the flowchart, determine the correctness of the given statement. Answer with either "true" or "false".*
	Question	There is an arrow between pennames vesuviate and monotrochal.	Ecological Systems can be influenced by itself.
	Answer	FALSE	TRUE
Description	Prompt	*The image contains a flowchart. Generate the description of the flowchart, reflecting the text nodes and arrows as depicted.*	
	Answer	pythagoreanly points to pennames vesuviate. pythagoreanly points to monotrochal. monotrochal points to pythagoreanly. pennames vesuviate points to pythagoreanly.	

Figure 7: Influence matrix schematic graph, based on [5, Figure 5]

Table 7.1: Common VQA tasks across both the Scientific and Simulated subsets of the FlowLearn Dataset.

7.2 FlowLearn Dataset

To address the scarcity of resources for flowchart comprehension, we introduce the FlowLearn Dataset. This dataset comprises two distinct subsets: Scientific Flowcharts and Simulated Flowcharts. An overview of the FlowLearn dataset, illustrating its components, is depicted in Figure 7.1. Table 7.1 details the common Visual Question Answering (VQA) tasks applicable to both subsets, while Table 7.2 lists the VQA tasks that are unique to the Simulated Flowcharts subset.

7.2.1 Scientific Flowchart Dataset

The Scientific Flowcharts Dataset comprises a comprehensive collection of flowchart images extracted from scholarly articles across diverse scientific domains. This dataset serves as a crucial resource for enhancing visual comprehension of scientific content.

7.2.1.1 Data Generation Process

We initiated our dataset creation by downloading 27,000 scientific articles from ArXiv. Using PDFFigures 2.0 [27], we extracted figures and related metadata. Additional insights were gained through the SciPDF Parser[1], which utilizes GROBID[2] for parsing PDFs.

Our selection process involved a rule-based filtering combined with manual annotation to identify flowcharts specifically. We excluded tables and figures placed

[1]https://github.com/titipata/scipdf_parser
[2]https://github.com/kermitt2/grobid

Task		Simulated Flowchart
Mermaid Code	Prompt	*The image contains a flowchart. Generate the Mermaid code to represent the flowchart, reflecting the text nodes and arrows as depicted.*

	Answer	```mermaid
		flowchart LR
		entity0(pythagoreanly)
		entity1(monotrochal)
		entity2(pennames vesuviate)
		entity0 ==>entity2
		entity0 –>entity1
		entity1 –>entity0
		entity2 –>entity0
		``

Num. of Nodes	Prompt	*The given image contains a simulated flowchart. You should find all text nodes and determine the total number of text nodes in the flowchart. Answer the question with a number.*
	Answer	3
Num. of Arrows	Prompt	*The given image contains a simulated flowchart. You should find all arrows and determine the total number of arrows in the flowchart. Answer the question with a number.*
	Answer	4

Table 7.2: VQA tasks unique to the Simulated Flowcharts subset of the FlowLearn Dataset.

at the end of papers (figure numbers above five), and selected figures based on keywords relevant to flowcharts in captions, such as "illustration," "flowchart," "model," "step," "overall," and "Graphical representation." Figures with unrelated keywords like "Normalized" and "Plot" were omitted. This meticulous curation yielded 3,858 flowcharts from 2,674 documents, focusing on images that prominently feature arrows, indicative of flowchart structures.

7.2.1.2 Dataset Contents

Each flowchart in this dataset is paired with extensive metadata and annotations. This includes Scientific Paper Meta containing data and parsed text from the source articles, enriching the context for each flowchart. Additionally, the Figure Meta details the captions and precise in-text mentions within the documents, allowing users to access and utilize image description text derived directly from the original sources. These captions are valuable for flowchart comprehension subtasks, such as image captioning and summarization. Furthermore, to support OCR subtask, we have annotated all text appearing in each flowchart using PaddleOCR [129].

7.2.2 Simulated Flowcharts

Recognizing that understanding flowcharts extends beyond caption generation, we developed the Simulated Flowcharts subset to enhance comprehension of diagrammatic components like arrows and nodes, which can be labor-intensive to annotate in scientific diagrams.

7.2.2.1 Dataset Generation Process

This subset was generated using Mermaid, a JavaScript tool that translates Markdown-inspired text definitions into flowcharts. Sample Mermaid code can be seen in Figure 7.1 and Table 7.2. We utilized Python scripts to introduce variability in the flowchart definitions in terms of the following aspects:

- **Nodes:** Each flowchart contains between 3 to 10 nodes, with node text consisting of randomized English words.

- **Links:** The number of links between nodes is randomized, with all nodes connected by at least one link, mimicking real-world flowchart structures. We randomize the type of arrow links between nodes, including solid lines, bold lines, or dashed lines.
- **Background Color and Flowchart Direction:** Both are randomized to add visual diversity and reflect different organizational styles.

7.2.2.2 Dataset Contents

We generated a total of 10,000 samples. Each sample includes:

- **Flowchart Images:** Available in JPEG and SVG formats to suit various usage scenarios.
- **Mermaid Code:** Provided for each sample to facilitate programmatic understanding and manipulation of the flowchart structure.
- **Visual Component Annotations:** Detailed annotations are provided, which include the node text and the precise locations of text nodes, arrowheads, and tails, all derived from SVG. These annotations are crucial for tasks such as object detection and structural analysis, enabling a deeper understanding of the flowchart components.

The generation script provides fine-grained control over the creation of simulated flowchart samples, enabling integrated training and experimentation for a wide range of applications.

7.2.3 Visual Question Answering

To evaluate the flowchart understanding capabilities using the FlowLearn dataset, we developed tailored Visual Question Answering (VQA) question-answer pairs for each tested flowchart. Examples of prompts, questions and answers for each task are detailed in Table 7.1 and Table 7.2. We have ensured that all prompts are elaborately detailed based on findings from VL-ICL[57], which demonstrated that more detailed prompts significantly enhance VQA performance compared to shorter ones. Our own experiments confirm this finding, as we observed that detailed prompts consistently outperform shorter, more concise ones in eliciting accurate responses from models.

7.2.3.1 Common Tasks

The common VQA tasks for both subsets include:

OCR: We randomly place a red box over one of the annotated texts within the flowchart and prompt models to identify and return the enclosed words.

True/False: We generate statements related to the flowchart and query the model to determine their veracity. For *Scientific Flowcharts*, we initially create two accurate statements using sentences from the figure caption, subsequently verified by annotators for their correctness based on the flowchart. In cases with insufficient caption data, annotators generate additional statements directly relating to the flowchart. For false statements, annotators alter a few words in a true statement to reverse its meaning, ensuring the vocabulary remains consistent with the original

author's style. This process yields one true and one false statement for each tested scientific flowchart.

For *Simulated Flowcharts*, we use predefined templates to create True and False statements, such as: "An arrow exists between node'{a}' and node'{b}'" and "An arrow points from node'{a}' to node'{b}'." where {a} and {b} are placeholders for node texts identified in Visual Component Annotations (Section 7.2.2.2).

Description: We prompt models to generate descriptions for the flowcharts. For *scientific flowcharts*, the reference answers are derived from their captions; for *simulated flowcharts*, reference answers are generated by converting mermaid code to sentences using templates, "{a} points to {b}."

7.2.3.2 Simulated Flowchart Tasks

Additionally, there are three tasks unique to the Simulated Flowchart subset:

Mermaid Code: Models are tasked with generating Mermaid code that represents the flowchart. This task assesses the model's ability to comprehensively recognize flowchart components, including text nodes and arrows.

Number of Nodes and Arrows: Models answer questions regarding the count of text nodes and arrows present in the flowchart. This task offers a quantitative measure of the model's comprehension, though it is less comprehensive than the Mermaid Code task.

7.3 Experiment Setups

In this section, we detail the experimental setup used to assess the capabilities of various Large Vision-Language Models (LVLMs) using the FlowLearn Dataset. Our primary objective is to evaluate how effectively these models comprehend and interpret flowcharts from both the Scientific and Simulated subsets. We have implemented all Visual Question Answering (VQA) tasks outlined in Section 7.2.3, which probe various facets of flowchart comprehension—from fundamental text recognition to more advanced logical reasoning and structural analysis.

7.3.1 Models

We selected Large Vision-Language Models (LVLMs) for evaluation based on their rankings in the OpenCampass multi-modal leaderboard as of April 2024. Access to some models was facilitated through APIs, including Step-1V-32K, GPT-4V, Gemini-Pro-Vision, and Claude-3-Opus-20240229. Additional models assessed in our study were LLaVA-v1.6-Vicuna-7B, InternLM-XComposer2-VL-7B, Qwen-VL-Chat from 2024/01/25, and DeepSeek-VL-7B-chat. Our selection strategy aimed to choose the best model available from each top-ranked model family, such as selecting Claude-3-Opus from the Claude series. However, we were unable to evaluate the top-ranked version of LLaVA on OpenCampass due to resource limitations.

7.3.2 Evaluation Metrics

To accurately gauge the performance of the models, we categorized the VQA tasks into three groups, each assessed by tailored evaluation metrics:

Accuracy: We measure the accuracy for tasks including OCR, True/False Statements, Number of Nodes, and Number of Arrows. This metric is straightforward and evaluates whether the responses are correct or incorrect based on the ground truth. Specifically for True/False Statements, we calculated average accuracy separately for the true and false subsets, and an overall average accuracy to provide a comprehensive view of model performance

Similarity: For description tasks, we assess the closeness of model-generated descriptions to reference descriptions using similarity metrics. Specifically, we utilize two metrics. First, the BERT score [130] employs pre-trained contextual embeddings from BERT to evaluate the semantic coherence between the model's response and the reference sentences. It achieves this by matching words through cosine similarity. Second, the Sentence Transformer [131] converts responses and reference sentences into embeddings using the 'all-MiniLM-L6-v2' model. We then employ cosine similarity to quantitatively determine the similarity, providing a precise measure of how closely the generated text aligns with the target description.

Mermaid Code Generation: We developed two sets of metrics specifically tailored for evaluating the correctness of generated Mermaid code:

- *Node-Level Evaluation:* This metric checks if all nodes present in the ground truth are included in the model's response. Each node is only considered correct if it exactly matches the spelling in the ground truth.
- *Link-Level Evaluation:* This metric assesses the generated response includes all the links present in the ground truth. A link is deemed correct if both the start and end nodes are accurately predicted, regardless of the arrow type. We also permit some syntactical flexibility in how node descriptions are expressed, allowing the use of either the node variable name or the node text.

For both levels of evaluation, we compute F1-score, precision, and recall for each sample and subsequently average these metrics across all samples.

7.3.3 Response Parsing

Given the variability in how LVLMs generate responses, which may not always exactly match the ground truth even when correct, we have developed specific rules to parse and evaluate the responses:

- **OCR:** A prediction is deemed correct if it includes the exact phrase from the ground truth.
- **True/False Statements:** The response is evaluated based on the presence of'true' or'false' within the statement, irrespective of case. If neither token is found, the response is marked as none.
- **Number of Nodes or Arrows:** We extract the first numeric token in the response, also converting English words representing numbers into numeric tokens. If no such token appears, the response is marked as none.

- **Mermaid Code Prediction:** We focus on statements encapsulated within triple backticks (```) in model responses. From these, we extract nodes and links according to the Mermaid syntax rules.

7.3.4 Settings

For our evaluations, we utilized the testing subset of the FlowLearn dataset, which included assessments of 500 scientific flowcharts and 2,000 simulated flowcharts. Due to cost constraints and API limitations, we limited our evaluations to 100 samples per task for Claude-3-Opus, GPT-4V, and Step-1V. All other evaluations were conducted using an NVIDIA A100 80GB GPU.

We opted for few-shot prompting as our evaluation strategy to align the output of the Large Vision-Language Models (LVLMs) more closely with the ground truth. According to vlicl, few-shot prompting, particularly with 2-shot samples, generally yields the most significant performance improvement in general vision-language VQA tasks across various LVLMs. Additionally, using 2-shot samples provides a balanced approach for evaluating True/False statements, as it allows an equal representation of both true and false scenarios within the prompts. This method ensures that the models are not biased toward one answer type over the other, facilitating a more accurate and fair assessment of model capabilities.

For consistency, we employ the prompt format shown in Table 7.3 for evaluation.

Prompt: [Task Description]
Support Set: [Image][Question][Answer] (2-shot)
Query: [Image][Question]
Prediction: [Answer]

Table 7.3: 2-Shot prompt format used for evaluation.

7.4 Experiment Results

In this section, we present the results from our evaluation of the Large Vision-Language Models (LVLMs) across three distinct groups of Visual Question Answering (VQA) tasks within the FlowLearn dataset. Each task group was designed to test different aspects of model performance using specialized evaluation metrics. For a focused review of performance across a limited subset of 100 samples involving all models and all tasks, please refer to Section 1 of the Supplementary Materials. The findings there align closely with the results discussed here. Sample model responses to all VQA tasks are shown in Section 2 of the Supplementary Materials.

Task		Claude	GPT4V	Step-1V	Gemini ProVision	InternLM -XComposer2-VL	LLaVA 16-7B	Qwen-VL -chat	DeepSeek -VL-7B-chat
Scientific Flowchart									
	OCR	0.44	0.51	**0.66**	0.43	0.06	0.04	0.08	0.05
Statements	TRUE	0.69	0.63	**0.9**	0.55	0.89	0.08	0.18	0.86
	FALSE	0.53	**0.74**	0.36	0.7	0.28	0.09	0	0.21
	Average	0.61	**0.68**	0.63	0.62	0.59	0.09	0.09	0.53
Simulated Flowchart									
	OCR	**0.83**	0.75	0.71	0.69	0.18	0	0.58	0.23
	Num. Nodes	0.52	**0.58**	0.31	0.02	0.15	0	0.4	0.22
	Num. Arrows	0.23	**0.26**	**0.26**	0.09	0.12	0.06	0.2	0.15
Statements: Between AB	TRUE	0.42	0.28	0.62	0.69	**0.82**	0.02	0.34	0.61
	FALSE	0.71	0.77	0.72	0.52	0.5	**0.91**	0	0.56
	Average	0.56	0.52	**0.67**	0.61	0.66	0.46	0.17	0.59
Statements: A to B	TRUE	0.18	0.15	0.41	0.16	**0.85**	0	0.04	0.55
	FALSE	0.49	0.61	0.5	0.63	0.49	**0.9**	0.01	0.58
	Average	0.34	0.38	0.45	0.4	**0.67**	0.45	0.03	0.57

Table 7.4: Experiment results for accuracy tasks. Models† are evaluated on a subset of the evaluation set. Regardless of evaluation size, the best-performing model is **bolded**. The best-performing model among those evaluated on the full set is underlined.

7.4.1 Accuracy Tasks

The first group of tasks evaluates the accuracy of the LVLMs in responding to queries that require precise, binary, or short phrase answers. These tasks are foundational for assessing flowchart comprehension. The performance of each model on these accuracy tasks is summarized in Table 7.4, leading to several key observations:

1) **No clear winner across all accuracy tasks.** For scientific flowcharts, Gemini-Pro-Vision showed the strongest performance on the full test set. However, on smaller subsets, GPT-4V and Step-1V also demonstrated strong performances. For simulated flowcharts, on the full test set, InternLM excelled in True/False statements, Gemini-Pro in OCR tasks, and Qwen-VL in counting nodes and arrows.

2) **Irrelevant model responses.** Although most models generally produced task-related responses, irrelevant responses were still observed. For True/False tasks, Qwen-VL often scored close to zero, indicating a lack of 'true' or 'false' tokens in its responses. Conversely, LLaVA frequently misclassified statements in simulated flowcharts as False, resulting in high scores for the false subset and negligible scores for the true subset.

3) **Challenges in counting nodes and arrows.** Counting tasks, which require comprehensive image understanding rather than partial recognition, proved difficult for most models, leading to lower average scores. Notably, despite its underperformance in other areas, Qwen-VL's results were comparatively better in these tasks.

Evaluation Metrics	Claude	GPT4V	Step-1V	Gemini ProVision	InternLM -XComposer2-VL	LLaVA 16-7B	Qwen-VL -chat	DeepSeek -VL-7B-chat
Scientific Flowchart								
BERTScore-F1	0.84	0.83	0.84	0.83	0.83	0.81	0.79	**0.86**
Sent.Transformer Similarity	**0.49**	0.46	0.42	0.30	0.34	0.25	0.38	0.36
Simulated Flowchart								
BERTScore-F1	0.90	0.90	0.89	**0.92**	0.82	0.77	0.80	0.87
Sent.Transformer Similarity	0.84	0.84	0.84	**0.88**	0.41	0.18	0.51	0.71

Table 7.5: Experiment results for Flowchart Description task. Models† are evaluated on a subset of the evaluation set. Regardless of evaluation size, the best-performing model is **bolded**. The best-performing model among those evaluated on the full set is underlined.

7.4.2 Similarity Tasks (Description)

The second group of tasks evaluates the ability of LVLMs to generate accurate descriptions of flowcharts. The performance of each model on the description task is detailed in Table 7.5. For the full set of scientific flowcharts, DeepSeek achieved the highest BERT Score, while Qwen-VL recorded the highest Sentence Transformer Similarity score. Regardless of the evaluation size, Claude, GPT-4V, and Step-1V also demonstrated strong performance across both metrics.

In the simulated flowchart evaluations, Gemini outperformed other models in both evaluation metrics. Typically, scores for simulated flowcharts were higher than those for scientific flowcharts. This difference is likely due to the structured nature of the reference answers for simulated flowcharts, which are generated using a consistent template, as opposed to the more varied language found in scientific flowchart captions. Additionally, scientific flowchart captions often contain extra contextual information not directly discernible from the flowchart itself, posing a challenge for models tasked with generating similar descriptions. This additional information can

Evaluation	Metric	Claude	GPT4V	Step-1V	Gemini ProVision	Gemini ProVision (CoT)	InternLM -XComposer2-VL	LLaVA 16-7B	Qwen-VL -chat	DeepSeek -VL-7B-chat
Link	Precision	**0.35**	0.23	0.14	0.26	0.25	0.01	0	0.02	0.05
	Recall	**0.26**	0.22	0.15	0.25	0.24	0.02	0	0.02	0.04
	F1	**0.3**	0.22	0.14	0.25	0.25	0.02	0	0.02	0.04
Node	Precision	**0.94**	0.72	0.68	0.75	0.71	0.09	0	0.06	0.29
	Recall	**0.95**	0.73	0.68	0.75	0.71	0.16	0	0.08	0.29
	F1	**0.94**	0.72	0.68	0.75	0.71	0.12	0	0.07	0.28

Table 7.6: Experiment results for Flowchart-to-Mermaid on Simulalted Flowcharts. Models† are evaluated on a subset of the evaluation set. Regardless of evaluation size, the best-performing model is **bolded**. The best-performing model among those evaluated on the full set is underlined.

skew the models' performance, making it difficult to achieve high similarity scores when compared to the original captions.

7.4.3 Mermaid Code Task

This task assesses the comprehensive ability of LVLMs to sencapsulate their understanding of a flowchart in a code format, summarizing aspects such as OCR, counting nodes and arrows, and recognizing relationships between nodes. The performance of each model on the Mermaid Code task for simulated flowcharts is summarized in Table 7.6. In evaluations on the full dataset, Gemini achieved the highest scores across all metrics. On a smaller evaluation subset, Claude demonstrated superior performance, particularly excelling in node-level prediction with an F1 score of 94%.

Challenges were notable in models like InternLM, LLaVA, Qwen-VL, and DeepSeek, all of which recorded scores close to zero. Several issues were identified during the evaluation of their outputs:

- **Syntax Compliance:** These models did not adhere to the proper syntax of Mermaid code, failing to correct their outputs even after 2-shot prompting designed to teach them the correct code format.
- **Node Recognition:** Disregarding syntax issues, these models still struggled to accurately predict correct nodes. The node-level evaluation, which also indirectly assesses models' OCR capabilities by checking for the presence of all node text in the predictions, reflected poor performance. This aligns with results from Table 7.4, where these models underperformed in OCR tasks that required text detection within specified areas.

Link-level predictions, which depend on accurate node-level results, consider a prediction correct only if the start and end nodes and the direction of the link are identified accurately. Consequently, scores for link-level evaluations generally fall below those for node-level evaluations. Even Claude, which scored highly at the node level, encountered significant challenges with link prediction, achieving only a 30% F1 score for link-level accuracy. This highlights the difficulty models face in understanding complex relationships within flowcharts.

7.4.4 Ablation Study on Chain-of-Thought

For complex tasks such as converting a flowchart into Mermaid code, a methodical approach can be beneficial. This process typically involves several sequential steps: initially detecting text nodes, then recognizing the links between them, and finally compiling these information into a standardized format, such as Mermaid code. Given the multi-step nature of this task, we hypothesized that introducing a chain-of-

First, the flowchart includes the following nodes: **{1}** Then, it contains the following edges: **{2}** Finally, the Mermaid code for the flowchart is: **{3}**

Table 7.7: Chain-of-Thought answer template.

thought(CoT) process could potentially enhance model performance. Consequently, we conducted an experimental ablation study on the simulated subset using Gemini-Pro-Vision, a model that incurs no querying cost and can be evaluated on the full test set. Notably, this model has shown the best performance on the Mermaid code task (Section 7.4.3).

For this experiment, we modified the 2-shot example answers using a structured template (Table 7.7) that guides the model through a step-by-step reasoning process. In this template, {1} is replaced with all text appearing in the simulated flowchart, {2} is derived from the flowchart description generated as per the templates described in Section 7.2.3.1, and {3} is the corresponding Mermaid code. Additionally, we appended the phrase "Let's think step by step" at the end of the original prompt (as illustrated in Table 7.2) to further emphasize the sequential reasoning process.

Surprisingly, the CoT performance, as shown in Table 7.6, indicated a slight decrease in compared to the original model configuration without the chain-of-thought. This unexpected outcome suggests that while the chain-of-thought method is intended to foster clearer and more structured reasoning, it may introduce additional complexities or dependencies that hinder the model's ability to synthesize and process information efficiently. Further analysis and refinement of

the implementation approach for the chain-of-thought may be necessary to capitalize on its potential benefits and overcome these challenges.

7.5 Discussion

As the initial version of the FlowLearn dataset, certain limitations are inherent, which provides opportunities for future enhancement and refinement.

7.5.1 Scientific Flowchart Subset Considerations

First, True/False statements are missing for the training set. In the scientific flowchart dataset, not all samples include related True/False statements. Annotators were tasked only with generating these statements for the test samples, leaving the remaining entries without this specific type of question-answer pair. Producing these statements is labor-intensive, with annotators spending an average of three minutes to verify and create each pair. Future versions could expand this task to cover all entries in the dataset.

Second, the dataset size is currently limited. With fewer than 4,000 images, the number of scientific flowcharts in FlowLearn is relatively modest compared to other common visual-language datasets that often contain millions of images. While the inclusion of simulated flowcharts helps to mitigate this limitation by broadening the scope of the training data, expanding the collection of scientific flowcharts would be advantageous. More scientific context images would enhance training for LVLMs by providing a richer array of real-world examples. Future versions could expand the size.

Third, the descriptive task is limited. The descriptive task for scientific flowcharts is currently evaluated against figure captions. However, the descriptive text for scientific diagrams is often scattered throughout the associated literature, as outlined in Context24[3]: Contextualizing Scientific Figures and Tables. A more robust approach would involve annotators extracting and collating descriptive text from the full text of scientific articles to provide a more comprehensive base for evaluating LVLM-generated descriptions.

7.5.2 Simulated Flowchart Subset Considerations

The simulated flowchart subset was designed to augment the scientific subset by offering a more granular evaluation of flowchart comprehension and providing additional training data. Future iterations could improve upon this by incorporating a greater diversity of diagram types, such as state diagrams and quadrant charts, to enrich the dataset further. While FlowLearn currently focuses exclusively on flowcharts, expanding the range of diagram types could enhance its applicability.

7.5.3 Model Selection

Our model selection was biased towards LVLMs due to their broad capabilities and general applicability. However, many task-specific, smaller visual-language models may also be well-suited for these tasks. Future work will explore the potential of these models, which might offer more specialized insights or efficiencies in specific aspects of flowchart comprehension.

[3]https://sdproc.org/2024/sharedtasks.html#context24

7.6 Conclusion

In this study, we introduced and evaluated the FlowLearn dataset, a novel resource aimed at advancing the understanding of flowcharts for visual-language models. Our experiments spanned various tasks, including OCR, True/False assessments, counting nodes and arrows, flowchart description, and generating Mermaid code, across two distinct subsets: scientific and simulated flowcharts.

Our findings demonstrate that while LVLMs are capable of impressive performance on certain tasks, challenges remain. Notably, the models excelled at OCR and True/False statements in certain contexts but struggled with the more complex task of accurately generating Mermaid code from flowcharts. This underscores a broader issue: LVLMs often struggle to fully comprehend the intricate relationships between visual components and to synthesize this information into structured code formats effectively.

Given the rapid advancements in the fields of LLMs and LVLMs, the FlowLearn dataset is timely and provides valuable insights into a relatively underexplored area. It not only serves as a critical tool for benchmarking and refining these models but also helps illuminate the specific difficulties they encounter with visual reasoning in a structured context. By pushing the boundaries of what LVLMs can understand and achieve, we can bridge the gap between human and machine comprehension of visual and language tasks, paving the way for more intelligent and capable automated systems.

BIBLIOGRAPHY

[1] Joseph Lau, Elliott M. Antman, Jeanette Jimenez-Silva, Bruce Kupelnick, Frederick Mosteller, and Thomas C. Chalmers. Cumulative meta-analysis of therapeutic trials for myocardial infarction. *New England Journal of Medicine*, 327(4):248–254, 1992.

[2] Qingyu Chen, Alexis Allot, and Zhiyong Lu. Litcovid: an open database of covid-19 literature. *Nucleic Acids Research*, 49(D1):D1534–D1540, 11 2020.

[3] Jader Martins Camboim de Sá, Marcos Da Silveira, and Cédric Pruski. Survey in characterization of semantic change, 2024.

[4] Wenhao Sun and Nicholas David. A critical reflection on attempts to machine-learn materials synthesis insights from text-mined literature recipes. *Faraday Discuss.*, 256:614–638, 2025.

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

[6] Huitong Pan, Qi Zhang, Cornelia Caragea, Eduard Dragut, and Longin Jan Latecki. SciDMT: A large-scale corpus for detecting scientific mentions. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14407–14417, Torino, Italia, May 2024. ELRA and ICCL.

[7] Huitong Pan, Qi Zhang, Cornelia Caragea, Eduard Dragut, and Longin Jan Latecki. Flowlearn: Evaluating large vision-language models on flowchart understanding, 2024.

[8] Yan Zhuang, Junyan Zhang, Ruogu Lu, Kunlun He, and Xiuxing Li. Medner: Enhanced named entity recognition in medical corpus via optimized balanced and deep active learning. *ACM Trans. Intell. Syst. Technol.*, 15(5), October 2024.

[9] Geraint Duck, Goran Nenadic, Andy Brass, David L Robertson, and Robert Stevens. bionerds: exploring bioinformatics' database and software use through literature mining. *BMC Bioinformatics*, 14(1), Jun 2013.

[10] Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. SemEval 2017 task 10: ScienceIE - extracting keyphrases

and relations from scientific publications. In *SemEval*, pages 546–555, August 2017.

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

[12] Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly. Identification of tasks, datasets, evaluation metrics, and numeric scores for scientific leaderboards construction. In *ACL*, pages 5203–5213, Florence, Italy, July 2019. Association for Computational Linguistics.

[13] Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *EMNLP*, pages 3219–3232, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[14] Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. SciREX: A challenge dataset for document-level information extraction. In *ACL*, pages 7506–7516, Online, July 2020. Association for Computational Linguistics.

[15] Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly. TDMSci: A specialized corpus for scientific literature entity tagging of tasks datasets and metrics. In *ACL*, pages 707–714, Online, April 2021. Association for Computational Linguistics.

[16] Prashant Singh, Erik Lehmann, and Mark Tyrrell. Climate policy transformer: Utilizing NLP to track the coherence of climate policy documents in the context of the Paris agreement. In Dominik Stammbach, Jingwei Ni, Tobias Schimanski, Kalyan Dutia, Alok Singh, Julia Bingler, Christophe Christiaen, Neetu Kushwaha, Veruska Muccione, Saeid A. Vaghefi, and Markus Leippold, editors, *Proceedings of the 1st Workshop on Natural Language Processing Meets Climate Change (ClimateNLP 2024)*, pages 1–11, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[17] Shijia Zhou, Siyao Peng, and Barbara Plank. CLIMATELI: Evaluating entity linking on climate change data. In Dominik Stammbach, Jingwei Ni, Tobias Schimanski, Kalyan Dutia, Alok Singh, Julia Bingler, Christophe Christiaen, Neetu Kushwaha, Veruska Muccione, Saeid A. Vaghefi, and Markus Leippold, editors, *Proceedings of the 1st Workshop on Natural Language Processing Meets Climate Change (ClimateNLP 2024)*, pages 215–222, Bangkok, Thailand,

August 2024. Association for Computational Linguistics.

[18] Ni Li, Shorouq Zahra, Mariana Brito, Clare Flynn, Olof Görnerup, Koffi Worou, Murathan Kurfali, Chanjuan Meng, Wim Thiery, Jakob Zscheischler, Gabriele Messori, and Joakim Nivre. Using LLMs to build a database of climate extreme impacts. In Dominik Stammbach, Jingwei Ni, Tobias Schimanski, Kalyan Dutia, Alok Singh, Julia Bingler, Christophe Christiaen, Neetu Kushwaha, Veruska Muccione, Saeid A. Vaghefi, and Markus Leippold, editors, *Proceedings of the 1st Workshop on Natural Language Processing Meets Climate Change (ClimateNLP 2024)*, pages 93–110, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[19] Aida Usmanova and Ricardo Usbeck. Structuring sustainability reports for environmental standards with LLMs guided by ontology. In Dominik Stammbach, Jingwei Ni, Tobias Schimanski, Kalyan Dutia, Alok Singh, Julia Bingler, Christophe Christiaen, Neetu Kushwaha, Veruska Muccione, Saeid A. Vaghefi, and Markus Leippold, editors, *Proceedings of the 1st Workshop on Natural Language Processing Meets Climate Change (ClimateNLP 2024)*, pages 168–177, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[20] Dario Garigliotti. SDG target detection in environmental reports using retrieval-augmented generation with LLMs. In Dominik Stammbach, Jingwei Ni, Tobias Schimanski, Kalyan Dutia, Alok Singh, Julia Bingler, Christophe Christiaen, Neetu Kushwaha, Veruska Muccione, Saeid A. Vaghefi, and Markus Leippold, editors, *Proceedings of the 1st Workshop on Natural Language Processing Meets Climate Change (ClimateNLP 2024)*, pages 241–250, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[21] Robert G. Raskin and Michael J. Pan. Knowledge representation in the semantic web for earth and environmental terminology (sweet). *Computers Geosciences*, 31(9):1119–1125, 2005. Application of XML in the Geosciences.

[22] Karl E Taylor, Martin Juckes, V Balaji, Luca Cinquini, Sébastien Denvil, Paul J Durack, Mark Elkington, Eric Guilyardi, Slava Kharin, Michael Lautenschlager, et al. Cmip6 global attributes, drs, filenames, directory structure, and cv's. *PCMDI Document*, 2018.

[23] D. Waliser, P. J. Gleckler, R. Ferraro, K. E. Taylor, S. Ames, J. Biard, M. G. Bosilovich, O. Brown, H. Chepfer, L. Cinquini, P. J. Durack, V. Eyring, P.-P. Mathieu, T. Lee, S. Pinnock, G. L. Potter, M. Rixen, R. Saunders, J. Schulz, J.-N. Thépaut, and M. Tuma. Observations for model intercomparison project (obs4mips): status for cmip6. *Geoscientific Model Development*, 13(7):2945–2958, 2020.

[24] Pengyuan Li, Xiangying Jiang, and Hagit Shatkay. Extracting figures and captions from scientific publications. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 1595–1598, New York, NY, USA, 2018. Association for Computing Machinery.

[25] Jian Wu, Jason Killian, Huaiyu Yang, Kyle Williams, Sagnik Ray Choudhury, Suppawong Tuarob, Cornelia Caragea, and C. Lee Giles. Pdfmef: A multi-entity knowledge extraction framework for scholarly documents and semantic search. In *Proceedings of the 8th International Conference on Knowledge Capture*, K-CAP 2015, New York, NY, USA, 2015. Association for Computing Machinery.

[26] Christopher Clark and Santosh Kumar Divvala. Looking beyond text: Extracting figures, tables and captions from computer science papers. In *AAAI Workshop: Scholarly Big Data*, 2015.

[27] Christopher Clark and Santosh Divvala. Pdffigures 2.0: Mining figures from research papers. 2016.

[28] Jian Chen, Meng Ling, Rui Li, Petra Isenberg, Tobias Isenberg, Michael Sedlmair, Torsten Möller, Robert S. Laramee, Han-Wei Shen, Katharina Wünsche, and Qiru Wang. Vis30k: A collection of figures and tables from ieee visualization conference publications. *IEEE Transactions on Visualization and Computer Graphics*, 27(9):3826–3833, 2021.

[29] Zeba Karishma, Shaurya Rohatgi, Kavya Shrinivas Puranik, Jian Wu, and C. Lee Giles. Acl-fig: A dataset for scientific figure classification, 2023.

[30] K. V. Jobin, Ajoy Mondal, and C. V. Jawahar. Docfigure: A dataset for scientific document figure classification. *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, Sep 2019.

[31] Ting-Yao Hsu, C Lee Giles, and Ting-Hao Huang. SciCap: Generating captions for scientific figures. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3258–3264, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[32] Iqra Safder, Hafsa Batool, Raheem Sarwar, Farooq Zaman, Naif Radi Aljohani, Raheel Nawaz, Mohamed Gaber, and Saeed-Ul Hassan. Parsing auc result-figures in machine learning specific scholarly documents for semantically-enriched summarization. *Applied Artificial Intelligence*, 36(1):2004347, 2022.

[33] Shaowei Wang, Lingling Zhang, Xuan Luo, Yi Yang, Xin Hu, Tao Qin, and Jun Liu. Computer science diagram understanding with topology parsing. *ACM*

Trans. Knowl. Discov. Data, 16(6), jul 2022.

[34] Lingdong Shen, Qigqi, Kun Ding, Gaofeng Meng, and Shiming Xiang. Rethinking comprehensive benchmark for chart understanding: A perspective from scientific literature, 2024.

[35] Rujing Yao, Linlin Hou, Yingchun Ye, Ou Wu, Ji Zhang, and Jian Wu. Method and Dataset Mining in Scientific Papers. *arXiv e-prints*, page arXiv:1911.13096, November 2019.

[36] Hyungrok Kim, Kinam Park, and Sae Park. Rich context competition: Extracting research context and dataset usage information from scientific publications. In *ACL*, pages 5203–5213, Florence, Italy, 2019. Association for Computing Machinery.

[37] He Zhao, Zhunchen Luo, Chong Feng, Anqing Zheng, and Xiaopeng Liu. A context-based framework for modeling the role and function of on-line resource citations in scientific literature. In *EMNLP*, pages 5206–5215, Hong Kong, China, November 2019. Association for Computational Linguistics.

[38] da Silva Viviane, Torres, Rademaker Alexandre, Lionti Krystelle, Giro Ronaldo, Lima Geisa, Fiorini Sandro, Archanjo Marcelo, Carvalho Breno, W., Neumann Rodrigo, Souza Anaximandro, Souza João, Pedro, Valnisio Gabriela, de, Paz Carmen, Nilda, Cerqueira Renato, and Steiner Mathias. Automated, llm enabled extraction of synthesis details for reticular materials from scientific literature. *arXiv preprint arXiv:2411.03484*, 2024.

[39] Garcia Gabriel, Lino, Ribeiro Manesco João, Renato, Paiola Pedro, Henrique, Miranda Lucas, de Salvo Maria, Paola, and Papa João, Paulo. A review on scientific knowledge extraction using large language models in biomedical sciences. *arXiv preprint arXiv:2412.03531*, 2024.

[40] Lin Qiu, Yunxuan Xiao, Yanru Qu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. Dynamically fused graph network for multi-hop reasoning. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6140–6150, Florence, Italy, July 2019. Association for Computational Linguistics.

[41] Haoyi Wu and Kewei Tu. Probabilistic transformer: A probabilistic dependency model for contextual word representation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7613–7636, Toronto, Canada, July 2023. Association for Computational Linguistics.

[42] Mozhdeh Gheini, Tatiana Likhomanenko, Matthias Sperber, and Hendra Setiawan. Joint speech transcription and translation: Pseudo-labeling with out-of-distribution data. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7637–7650, Toronto, Canada, July 2023. Association for Computational Linguistics.

[43] Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. Lightrag: Simple and fast retrieval-augmented generation, 2024.

[44] Hanzhu Chen, Xu Shen, Qitan Lv, Jie Wang, Xiaoqi Ni, and Jieping Ye. Sac-kg: Exploiting large language models as skilled automatic constructors for domain knowledge graphs, 2024.

[45] Keven Ates and Kang Zhang. Constructing veggie: Machine learning for context-sensitive graph grammars. *19th IEEE International Conference on Tools with Artificial Intelligence(ICTAI 2007)*, 2:456–463, 2007.

[46] Xiang Wei, Yufeng Chen, Ning Cheng, Xingyu Cui, Jinan Xu, and Wenjuan Han. Collabkg: A learnable human-machine-cooperative information extraction toolkit for (event) knowledge graph construction, 2023.

[47] Feiliang Ren, Jiaqi Wang, Yuying Chang, and Zhong Li. Techgpt 2.0: Technology-oriented generative pretrained transformer 2.0. `https://github.com/neukg/TechGPT-2.0`, 2023.

[48] Xu Derong, Zhang Ziheng, Zhu Zhihong, Lin Zhenxi, Liu Qidong, Wu Xian, Xu Tong, Zhao Xiangyu, Zheng Yefeng, and Chen Enhong. Mitigating hallucinations of large language models in medical information extraction via contrastive decoding. *arXiv preprint arXiv:2410.15702*, 2024.

[49] Li Sihang, Huang Jin, Zhuang Jiaxi, Shi Yaorui, Cai Xiaochen, Xu Mingjun, Wang Xiang, Zhang Linfeng, Ke Guolin, and Cai Hengxing. Scilitllm: How to adapt llms for scientific literature understanding. *arXiv preprint arXiv:2408.15545*, 2024.

[50] Yue Ming, Nannan Hu, Chunxiao Fan, Fan Feng, Jiangwan Zhou, and Hui Yu. Visuals to text: A comprehensive review on automatic image captioning. *IEEE/CAA Journal of Automatica Sinica*, 9(8):1339–1365, 2022.

[51] Zanyar Zohourianshahzadi and Jugal Kumar Kalita. Neural attention for image captioning: review of outstanding methods. *Artificial Intelligence Review*, 55:3833 – 3862, 2021.

[52] Ali Mazraeh Farahani, Peyman Adibi, Mohammad Saeed Ehsani, Hans-Peter Hutter, and Alireza Darvishy. Automatic chart understanding: A review. *IEEE Access*, 11:76202–76221, 2023.

[53] Venkat Kodali and Daniel Berleant. Recent, rapid advancement in visual question answering: a review. *2022 IEEE International Conference on Electro Information Technology (eIT)*, pages 139–146, 2022.

[54] Anbara Z Al-Jamal, Maryam J Bani-Amer, and Shadi A. Aljawarneh. Image captioning techniques: A review. *2022 International Conference on Engineering & MIS (ICEMIS)*, pages 1–5, 2022.

[55] Yue Ming, Nannan Hu, Chunxiao Fan, Fan Feng, Jiang Zhou, and Hui Yu. Visuals to text: A comprehensive review on automatic image captioning. *IEEE/CAA Journal of Automatica Sinica*, 9:1339–1365, 2022.

[56] Zanyar Zohourianshahzadi and Jugal Kumar Kalita. Neural attention for image captioning: review of outstanding methods. *Artificial Intelligence Review*, 55:3833 – 3862, 2021.

[57] Yongshuo Zong, Ondrej Bohdal, and Timothy Hospedales. Vl-icl bench: The devil in the details of benchmarking multimodal in-context learning, 2024.

[58] Noah Siegel, Zachary Horvitz, Roie Levin, Santosh Divvala, and Ali Farhadi. Figureseer: Parsing result-figures in research papers. In *European Conference on Computer Vision (ECCV)*, 2016.

[59] Fuji Ren and Yangyang Zhou. Cgmvqa: A new classification and generative model for medical visual question answering. *IEEE Access*, 8:50626–50636, 2020.

[60] Medfusenet: An attention-based multimodal deep learning model for visual question answering in the medical domain. 2021.

[61] Yash Khare, Viraj Bagal, Minesh Mathew, Adithi Devi, U Deva Priyakumar, and CV Jawahar. Mmbert: Multimodal bert pretraining for improved medical vqa, 2021.

[62] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Joshua B. Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. In *Neural Information Processing Systems*, 2018.

[63] Hikaru Shindo, Viktor Pfanschilling, Devendra Singh Dhami, and Kristian Kersting. alphailp: thinking visual scenes as differentiable logic programs.

Machine Learning, 112(5):1465–1497, Mar 2023.

[64] Haotian Li, Yong Wang, Aoyu Wu, Huan Wei, and Huamin Qu. Structure-aware visualization retrieval. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery.

[65] Xinlu Zhang, Yujie Lu, Weizhi Wang, An Yan, Jun Yan, Lianke Qin, Heng Wang, Xifeng Yan, William Yang Wang, and Linda Ruth Petzold. Gpt-4v(ision) as a generalist evaluator for vision-language tasks, 2023.

[66] Chaoyou Fu, Renrui Zhang, Zihan Wang, Yubo Huang, Zhengye Zhang, Longtian Qiu, Gaoxiang Ye, Yunhang Shen, Mengdan Zhang, Peixian Chen, Sirui Zhao, Shaohui Lin, Deqiang Jiang, Di Yin, Peng Gao, Ke Li, Hongsheng Li, and Xing Sun. A challenger to gpt-4v? early explorations of gemini in visual expertise, 2023.

[67] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023.

[68] Anthropic. The claude 3 model family: Opus, sonnet, haiku.

[69] Pan Zhang, Xiaoyi Dong, Bin Wang, Yuhang Cao, Chao Xu, Linke Ouyang, Zhiyuan Zhao, Haodong Duan, Songyang Zhang, Shuangrui Ding, Wenwei Zhang, Hang Yan, Xinyue Zhang, Wei Li, Jingwen Li, Kai Chen, Conghui He, Xingcheng Zhang, Yu Qiao, Dahua Lin, and Jiaqi Wang. Internlm-xcomposer: A vision-language large model for advanced text-image comprehension and composition, 2023.

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

[71] Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. Deepseek-vl: Towards real-world vision-language understanding, 2024.

[72] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision, 2021.

[73] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers,

Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, Julian Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bošnjak, Xi Chen, Matthias Minderer, Paul Voigtlaender, Ioana Bica, Ivana Balazevic, Joan Puigcerver, Pinelopi Papalampidi, Olivier Henaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiaohua Zhai. Paligemma: A versatile 3b vlm for transfer, 2024.

[74] OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Madry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu,

Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech

Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. Gpt-4o system card, 2024.

[75] Yuntian Deng, Anssi Kanervisto, Jeffrey Ling, and Alexander M. Rush. Image-to-markup generation with coarse-to-fine attention. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 980–989. JMLR.org, 2017.

[76] Damien Masson, Sylvain Malacria, Daniel Vogel, Edward Lank, and Géry Casiez. Chartdetective: Easy and accurate interactive data extraction from complex vector charts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA, 2023. Association for Computing Machinery.

[77] Jorge-Ivan Herrera-Camara and Tracy Hammond. Flow2code: from hand-drawn flowcharts to code execution. In *Proceedings of the Symposium on Sketch-Based Interfaces and Modeling*, SBIM '17, New York, NY, USA, 2017. Association for Computing Machinery.

[78] Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. S2ORC: The semantic scholar open research corpus. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online, July 2020. Association for Computational Linguistics.

[79] Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

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

[81] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. brat: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations Session at EACL 2012*, Avignon, France, April 2012. Association for Computational Linguistics.

[82] Ii. I Ntroduction. The ace 2005 (ace 05) evaluation plan evaluation of the detection and recognition of ace entities, values, temporal expressions, relations, and events. 2005.

[83] Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of english: The penn treebank. *Comput. Linguist.*, 19(2):313–330, jun 1993.

[84] Lisa S. Pearl and Jon Sprouse. Syntactic islands and learning biases: Combining experimental syntax and computational modeling to investigate the language acquisition problem. *Language Acquisition*, 20:23 – 68, 2013.

[85] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[86] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. In *Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2019.

[87] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics.

[88] Tomas Mikolov, Kai Chen, Greg S. Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[89] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv.org*, 2020.

[90] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2356–2362, 2021.

[91] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. Colbertv2: Effective and efficient retrieval via lightweight late interaction. *arXiv preprint arXiv:2112.01488*, 2021.

[92] Michael Färber, Alexander Albers, and Felix Schüber. Identifying used methods and datasets in scientific publications. In *SDU@ AAAI*, 2021.

[93] Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 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*, pages 1003–1011, Suntec, Singapore, August 2009. Association for Computational

Linguistics.

[94] Muhammad Abdul-Mageed and Lyle Ungar. EmoNet: Fine-grained emotion detection with gated recurrent neural networks. In *Association for Computational Linguistics (ACL)*, pages 718–728, Vancouver, Canada, July 2017.

[95] Jumanah Alshehri, Marija Stanojevic, Eduard Dragut, and Zoran Obradovic. On label quality in class imbalance setting -a case study. In *ICMLA*, pages 1666–1671, 2022.

[96] Peng Su, Gang Li, Cathy Wu, and K. Vijay-Shanker. Using distant supervision to augment manually annotated data for relation extraction. *PLOS ONE*, 14(7):e0216913, Jul 2019.

[97] Shanshan Zhang, Lihong He, Slobodan Vucetic, and Eduard Dragut. Regular expression guided entity mention mining from noisy web data. In *EMNLP*, pages 1991–2000, 2018.

[98] Shanshan Zhang, Lihong He, Eduard Dragut, and Slobodan Vucetic. How to invest my time: Lessons from human-in-the-loop entity extraction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2305–2313, 2019.

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

[100] Kathrin Blagec, Adriano Barbosa-Silva, Simon Ott, and Matthias Samwald. A curated, ontology-based, large-scale knowledge graph of artificial intelligence tasks and benchmarks. *Scientific Data*, 9, 2021.

[101] Mark Davies and Joseph L. Fleiss. Measuring agreement for multinomial data. *Biometrics*, 38(4):1047–1051, 1982.

[102] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022.

[103] Bernal Jimenez Gutierrez, Nikolas McNeal, Clay Washington, You Chen, Lang Li, Huan Sun, and Yu Su. Thinking about gpt-3 in-context learning for biomedical ie? think again. *ArXiv*, abs/2203.08410, 2022.

[104] Omer Levy, Yoav Goldberg, and Ido Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225, Dec 2015.

[105] Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. A latent variable model approach to pmi-based word embeddings. *Transactions of the Association for Computational Linguistics*, 4:385–399, Dec 2016.

[106] Andrew Schneider, Lihong He, Zhijia Chen, Arjun Mukherjee, and Eduard Dragut. COIN – an inexpensive and strong baseline for predicting out of vocabulary word embeddings. In *COLING*, pages 3984–3993, October 2022.

[107] Eduard C. Dragut, Hong Wang, Clement T. Yu, A. Prasad Sistla, and Weiyi Meng. Polarity consistency checking for sentiment dictionaries. In *ACL*, pages 997–1005, 2012.

[108] Eduard C. Dragut, Hong Wang, A. Prasad Sistla, Clement T. Yu, and Weiyi Meng. Polarity consistency checking for domain independent sentiment dictionaries. *TKDE*, 27(3):838–851, 2015.

[109] Andrew T. Schneider and Eduard C. Dragut. Towards debugging sentiment lexicons. In *ACL*, pages 1024–1034, 2015.

[110] Alisa Smirnova and Philippe Cudré-Mauroux. Relation extraction using distant supervision. *ACM Computing Surveys*, 51(5):1–35, Nov 2018.

[111] Danielle Touma, Samantha Stevenson, Flavio Lehner, and Sloan Coats. Human-driven greenhouse gas and aerosol emissions cause distinct regional impacts on extreme fire weather. In *AGU Fall Meeting Abstracts*, volume 2021, pages A51E–01, December 2021.

[112] Guodong Fan, Boru Zhou, Chengwen Meng, Tengwei Pang, Xi Zhang, Mingshu Du, and Wei Zhao. Development of a comprehensive physics-based battery model and its multidimensional comparison with an equivalent-circuit model: Accuracy, complexity, and real-world performance under varying conditions, 2024.

[113] Rihao Chang, Yongtao Ma, Tong Hao, and Weizhi Nie. 3d shape knowledge graph for cross-domain 3d shape retrieval, 2023.

[114] Vineeth Venugopal, Sumit Pai, and Elsa Olivetti. Matkg: The largest knowledge graph in materials science – entities, relations, and link prediction through graph representation learning, 2022.

[115] Shimizu Cogan, Stephe Shirly, Barua Adrita, Cai Ling, Christou Antrea, Currier Kitty, Dalal Abhilekha, Fisher Colby, K., Hitzler Pascal, Janowicz Krzysztof, Li Wenwen, Liu Zilong, Mahdavinejad Mohammad, Saeid, Mai Gengchen, Rehberger Dean, Schildhauer Mark, Shi Meilin, Norouzi Sanaz, Saki, Tian Yuanyuan, Wang Sizhe, Wang Zhangyu, Zalewski Joseph, Zhou Lu, and Zhu Rui. The knowwheregraph ontology. *arXiv preprint arXiv:2410.13948*, 2024.

[116] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization, 2024.

[117] Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen. Large language models for generative information extraction: A survey, 2024.

[118] Lei Yu, Meng Cao, Jackie Chi Kit Cheung, and Yue Dong. Mechanistic understanding and mitigation of language model non-factual hallucinations, 2024.

[119] Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. GLiNER: Generalist model for named entity recognition using bidirectional transformer. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5364–5376, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[120] Kishan Nagendra, Omran A. Bukhres, Srinivasan Sikkupparbathyam, Marcelo Areal, Zina Ben-Miled, Lola M. Olsen, Chris Gokey, David Kendig, Tom Northcutt, Rosy Cordova, and Gene Major. Nasa global change master directory: an implementation of asynchronous management protocol in a heterogeneous distributed environment. *Proceedings 3rd International Symposium on Distributed Objects and Applications*, pages 136–145, 2001.

[121] Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*, 2024.

[122] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022.

[123] Bingsheng Yao, Guiming Chen, Ruishi Zou, Yuxuan Lu, Jiachen Li, Shao Zhang, Yisi Sang, Sijia Liu, James Hendler, and Dakuo Wang. More samples

or more prompts? exploring effective few-shot in-context learning for LLMs with in-context sampling. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1772–1790, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[124] Zhuyun Dai, Vincent Y. Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B. Hall, and Ming-Wei Chang. Promptagator: Few-shot dense retrieval from 8 examples, 2022.

[125] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic,

Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres,

Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024.

[126] DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang,

Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. Deepseek-v3 technical report, 2024.

[127] David Thulke, Yingbo Gao, Petrus Pelser, Rein Brune, Rricha Jalota, Floris Fok, Michael Ramos, Ian van Wyk, Abdallah Nasir, Hayden Goldstein, Taylor Tragemann, Katie Nguyen, Ariana Fowler, Andrew Stanco, Jon Gabriel, Jordan Taylor, Dean Moro, Evgenii Tsymbalov, Juliette de Waal, Evgeny Matusov, Mudar Yaghi, Mohammad Shihadah, Hermann Ney, Christian Dugast, Jonathan Dotan, and Daniel Erasmus. Climategpt: Towards ai synthesizing interdisciplinary research on climate change, 2024.

[128] Sergei Bogdanov, Alexandre Constantin, Timothée Bernard, Benoit Crabbé, and Etienne Bernard. Nuner: Entity recognition encoder pre-training via llm-annotated data, 2024.

[129] Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. Towards accurate scene text recognition with semantic reasoning networks. In *Proceedings of the IEEE/CVF conference on computer vision and*

pattern recognition, pages 12113–12122, 2020.

[130] Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020.

[131] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Conference on Empirical Methods in Natural Language Processing*, 2019.

APPENDIX A

PROMPT

Table A.1 shows the prompt being used for Climate Science Entity and Relationship Extraction from the climate science literature. Table A.2 shows the prompt template for refining the node definitions.

-Goal-

Given a text document with a preliminary list of potential entities, verify, and identify all entities of the specified types within the text. Note that the initial list may contain missing or incorrect entities. Additionally, determine and label the relationships among the verified entities.

-Entity Types-

A project refers to the scientific program, field campaign, or project from which the data were collected.

A location is a place on Earth, a location within Earth, a vertical location, or a location outside of the Earth.

A model is a sophisticated computer simulation that integrate physical, chemical, biological, and dynamical processes to represent and predict Earth's climate system.

An experiment is a structured simulation designed to test specific hypotheses, investigate climate processes, or assess the impact of various forcings on the climate system.

A platform refers to a system, theory, or phenomenon that accounts for its known or inferred properties and may be used for further study of its characteristics.

A instrument is a device used to measure, observe, or calculate.

A provider is an organization, an academic institution or a commercial company.

A variable is a quantity or a characteristic that can be measured or observed in climate experiments.

A weather event is a meteorological occurrence that impacts Earth's atmosphere and surface over short timescales.

A natural hazard is a phenomenon with the potential to cause significant harm to life, property, and the environment.

A teleconnection is a large-scale pattern of climate variability that links weather and climate phenomena across vast distances.

An ocean circulation is the large-scale movement of water masses in Earth's oceans, driven by wind, density differences, and the Coriolis effect, which regulates Earth's climate.

-Relationship Types and Definitions-

ComparedTo: The source entity is compared to the target entity. Outputs: A climate model, experiment, or project (source entity) outputs data (target entity).

RunBy: Experiments or scenarios (source entity) are run by a climate model (target entity).

ProvidedBy: A dataset, instrument, or model (source entity) is created or managed by an organization (target entity).

ValidatedBy: The accuracy or reliability of model simulations (source entity) is confirmed by datasets or analyses (target entity).

UsedIn: An entity, such as a model, simulation tool, experiment, or instrument (source entity), is utilized within a project (target entity).

MeasuredAt: A variable or parameter (source entity) is quantified or recorded at a geographic location (target entity).

MountedOn: An instrument or measurement device (source entity) is physically attached or installed on a platform (target entity).

TargetsLocation: An experiment, project, model, weather event, natural hazard, teleconnection, or ocean circulation (source entity) is designed to study, simulate, or focus on a specific geographic location (target entity).

-Steps-

1. Identify all entities. For each identified entity, extract the following information:

- entity name: Name of the entity

- entity type: One of the following types: [project, location, model, experiment, platform, instrument, provider, variable]

Format each entity as ("entity"¡—¿¡entity name¿¡—¿¡entity type¿¡—¿¡entity description¿)

2. From the entities identified from step 1, identify all pairs of (source entity, target entity) that are *clearly related* to each other.

For each pair of related entities, extract the following information:

- source entity: name of the source entity

- target entity: name of the target entity

- relationship type: One of the following relationship types: ComparedTo, Outputs, RunBy, ProvidedBy, ValidatedBy, UsedIn, MeasuredAt, MountedOn, TargetsLocation

Format each relationship as (relationship | <source entity> | <target entity> | <relationship type>)

3. Return output in English as a single list of all the entities and relationships identified in steps 1 and 2. Use ## as the list delimiter. Do not output any code or steps for solving the question.

4. When finished, output < |COMPLETE| >

######################

-Examples-

{formatted examples}

##########

-Real Data-

##########

Text: {input text}

Potential Entities: {potential entities}

##########

Output:

Table A.1: Prompt Template for Climate Science Entity and Relationship Extraction

Given the following metadata about an entity in a climate science ontology, which may include the entity's name, ontology path, and a definition (which may be missing), please develop an edited definition suitable for a named entity recognition (NER) task in climate science literature. The definition should be concise, clear, and limited to 150 tokens. Ensure it is precise and emphasizes the entity's unique aspects, avoiding overly general descriptions that could apply to multiple entities. Do not explain; only provide the edited definition.

Metadata: {}

Edited Definition:

Table A.2: Prompt Template for Refining Definitions