A Position Paper on the Automatic Generation of Machine Learning Leaderboards

Anonymous ACL submission

Abstract

An important task in machine learning (ML) research is comparing prior work, which is often performed via *ML leaderboards*: a tabular overview of experiments with comparable conditions (e.g. same task, dataset, and metric). However, the growing volume of literature creates challenges in creating and maintaining these leaderboards. To ease this burden, researchers have developed methods to extract *leaderboard entries* from research papers for automated leaderboard curation. Yet, prior work varies in problem framing, complicating comparisons and limiting real-world applicability. In this position paper, we present the **first overview of Automatic Leaderboard Generation (ALG) research**, identifying fundamental differences in assumptions, scope, and output formats. We propose an **ALG unified conceptual framework** to standardise how the ALG task is defined. We offer **ALG benchmarking guidelines**, including recommendations for datasets and metrics that promote fair, reproducible evaluation. Lastly, we outline **challenges and new directions for ALG**, advocating for broader coverage by including all reported results and richer metadata.

1 Introduction

In today's fast-paced Machine Learning (ML) research environment, keeping abreast of advancements is more crucial than ever. The exponential growth in publications, exemplified by nearly a quarter of a million arXiv submissions in 2024, underscores the expanding global community of scholars and the accelerating pace of research (arXiv, 2025). This vast increase in information presents researchers with both rich opportunities for discovery but also makes it increasingly difficult to stay up to date.

A key task for researchers is comparing past study outcomes to identify state-of-the-art results or benchmark against prior work. In ML, this is typically done using leaderboards: tables of experimental results under comparable conditions (e.g. *task, dataset, metric*). The popularity of platforms like Papers with Code[1] underscores their value in providing accessible, up-to-date comparisons that help researchers track progress and identify leading methods.

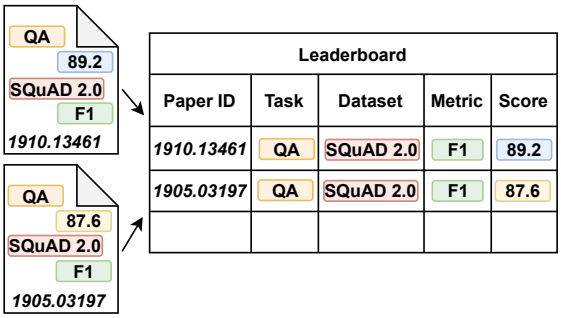

Figure 1: An example of extracting ⟨task, dataset, metric, score⟩ tuples from research papers to build a leaderboard[2].

However, leaderboards on these platforms are often incomplete or missing for certain tasks, and they typically rely on manual updates. To reduce this manual effort, recent work has focused on automatically extracting experimental outcomes (referred to here as "tuples") from research papers to populate leaderboards. We refer to this body of work as *Automatic Leaderboard Generation* (ALG): "A systematic process for extracting relevant experimental findings from scientific publications to create and maintain a leaderboard.". Figure 1 illustrates an example of this process, showing the extraction of ⟨*task, dataset, metric, score*⟩ tuples from two research papers to construct a leaderboard.

Research on ALG using natural language pro-

[1] https://paperswithcode.com

[2] An example of two SciLead (Şahinüç et al., 2024) leaderboard entries summarising Lewis et al. (2020) and Dong et al. (2019).

cessing (NLP) methodologies has seen significant developments in recent years. Indeed, there are still many open research questions as exemplified by the 2024 shared task on ALG (D'Souza et al., 2024), underscoring the ongoing relevance of ALG. This growing body of work has led to varied problem formulations and evaluation approaches, including differing assumptions about prior knowledge (§ 2.1) and extraction scope (§ 2.2), which makes comparisons across work difficult.

This position paper makes four important contributions. First, we provide the **first overview of ALG efforts** (§ 2-§ 4). By comparing prior studies side-by-side, we identify key divergences, such as variations in the assumed input scope (e.g. open vs. closed-domain) and captured results information, that previously hindered apples-to-apples comparisons. Our analysis provides a much-needed baseline map of the field, clarifying the field's current state and identifying critical gaps.

Second, based on this comparison, we propose an **ALG unified conceptual framework** (§ 5), essentially a problem formulation with unified terminology. This framework consolidates prior formulations into a coherent schema, providing a common language for researchers and enabling direct comparison of approaches.

Third, we provide **ALG Benchmarking Guidelines** (§ 6), to unify evaluation practices, addressing the previous lack of consensus. These guidelines establish shared standards for consistent, transparent evaluation and reliable progress tracking.

Fourth, we outline **challenges and new directions for ALG** (§ 7). We advocate expanding the extraction schema beyond just "best scores" to include all reported results (e.g. baselines, ablations) and enriching tuples with metadata (e.g. model architecture, hyperparameters) to enable more flexible result filtering.

Ultimately, the goal of this position paper is to resolve long-standing fragmentation, establish shared standards, and open new horizons for ALG.

2 Overview of Problem Definition

The ALG field has seen many advances over the years. At a broad level, the ALG task is an information extraction task, to extract a tuple containing key details of an ML experimental result.[3]

Hou et al. (2019) and Singh et al. (2019) laid the foundation by introducing methods for extracting leaderboard tuples directly from research papers. These methodologies have since been refined and expanded upon by new methods such as AxCell (Kardas et al., 2020), which was put into production by Papers with Code. The most recent methodologies use prompting of pre-trained Large Language Models (LLMs), e.g. prompting Llama 2 7B (Touvron et al., 2023) and Mistral 7B (Jiang et al., 2023) to extract ⟨task, dataset, metric, score⟩ tuples from research papers (Kabongo et al., 2024).

A key issue in the field is the variation in input and output expectations across studies. Table 1 lists key ALG papers we examined, focusing on recent work using transformer models that enable data scaling.[4]

We can characterise the key differences in the problem definition as concerning expectations about input and output data. Specifically, we discuss: (1) reliance on domain knowledge, and (2) limited scope of extraction.[5]

2.1 Reliance on Domain Knowledge

We observe that the ALG domains can be categorised as having different levels of reliance on prior domain knowledge, which ultimately impacts *what* information can be extracted. Essentially, two variants of the problem have been previously tackled: *closed domain* and *open domain*.[6]

Closed Domain: The closed-domain ALG problem stipulates that all the entities or tuples are predefined.[7] In the field, there have been two subvariants that we name: (1) *predefined typed entities* (PTE) and (2) *predefined typed tuples* (PTT).[8]

We define the *predefined typed entities* (PTE) as: "A closed-domain problem for ALG, in which the system is supplied with a finite catalogue of scientific concept classes (for instance, specific tasks, datasets, or metrics), and extractions are confined to items from that predefined list." The system may be given a declarative resource specifying entities,

[3] We acknowledge that ALG work rests on a long history of work in information extraction (IE) in scientific literature. The full body of IE work is out of scope for this analysis but is introduced briefly in Appendix A.

[4] Details on prior work are in Appendix C.

[5] We also note that various works have differed in expectations on the data format (e.g., PDF or LaTeX). However, we do not see this as critical in hindering comparisons of results.

[6] The "open domain" category includes hybrid cases that start with no domain knowledge and incrementally builds up knowledge as publications are processed.

[7] As in, bound by the closed world assumption.

[8] We borrow "predefined" from Şahinüç et al. (2024).

Methodology	Domain	Structured Data	Scope of Extraction
TDMS-IE Hou et al. (2019)	closed	Y	⟨task, dataset, metric⟩ & best score
PI Graph Singh et al. (2019)	open	Y	undefined
AxCell Kardas et al. (2020)	closed	Y	⟨task, dataset, metric⟩ & best score
SciREX-IE Jain et al. (2020)	open	Y	⟨task, dataset, metric, method⟩, no score
ORKG-TDM Kabongo et al. (2021)	closed	Y	⟨task, dataset, metric⟩, no score
TELIN Yang et al. (2022)	open	Y	⟨task, dataset, metric⟩, best score*
ORKG-LB Kabongo et al. (2023b)	closed	Y	⟨task, dataset, metric⟩, no score
TDMS-PR Kabongo et al. (2024)	open	Y	⟨task, dataset, metric⟩ & best score
MS-PR Singh et al. (2024)	open	N	⟨task⟩ & best score
TDMR-PR Şahinüç et al. (2024)	open	N	⟨task, dataset, metric⟩ & best score

* The scope of extraction is ambiguous (Yang et al., 2022). A response from the authors is pending for clarification.

Table 1: Characterisation of problem framing per method. Domain: open if extraction does not rely on prior knowledge, closed if restricted to a defined scope. Structured Data: Y if leaderboard tuples must appear in specific paper sections (e.g. tables or results), N otherwise. Scope of Extraction: extent of tuples extracted.

such as in Kardas et al. (2020). This could take the form of a taxonomy, a hierarchical structure of scientific concepts (e.g. tasks, datasets, metrics), or a simpler list of scientific named entities.

PTT is a further restriction beyond PTE in that only prescribed combinations of these science concepts are considered for establishing new tuples. We define PTT as "A closed-domain problem for ALG, in which a system is only allowed to detect leaderboard entries composed of specific, predefined combinations of known scientific concepts rather than forming any new combination." In PTT variants of ALG, only predetermined combinations (often observed combinations) are used for creating new tuples (e.g., as in Hou et al. (2019)).

Open Domain: An open-domain problem allows extraction of novel entities or tuples without relying on prior knowledge (e.g. taxonomies or lists), making it less constrained. This setup is often more application-friendly, as the extraction scope is guided solely by the user's information needs.

While more appealing to users, the open-domain variant requires handling duplicates, as the same concept may appear in different forms (e.g. "ROUGE" vs. "RGE" (Jain et al., 2020; Şahinüç et al., 2024)). This makes evaluation harder than in the closed domain, where canonical representations (e.g. predefined strings) enable direct accuracy measurement. Open-domain outputs may require fuzzy or semantic comparison metrics to handle variation.

2.2 Scope of Extraction

Beyond differences in domain knowledge, extraction scope also varies. Prior work differs in which classes of scientific concepts, typically methodological attributes like task, dataset, method, metric, and score, are included.

Furthermore, most work focuses only on extracting the top results from each paper, restricting each paper to a single entry per leaderboard (Hou et al., 2019; Kardas et al., 2020; Hou et al., 2021; Yang et al., 2022). If a publication presents two methods, only the top-performing one typically appears on the leaderboard. This can lead to an incomplete and potentially biased view, omitting valuable contributions such as negative results.[9]

3 Overview of ALG Datasets

With the growth of the field, several datasets have been proposed to evaluate ALG methods, making it hard for researchers to identify which datasets are best suited for benchmarking. To guide dataset selection, Table 2 summarises their key characteristics[10]. We highlight the main dimensions along which datasets differ. The main takeaway from this table is the diversity of the datasets that have been used in past research, making it hard to make fair comparisons. We discuss the variations below. A few recent datasets offer valuable attributes: LEGOBench (Singh et al., 2024) is the largest and covers the broadest tuple scope (including score), while SciLead (Şahinüç et al., 2024) stands out for its exhaustive manual annotations.

3.1 ML Experiment Science Entities

As prior work has varied in the entity classes studied, datasets have likewise differed in the scope of their tuple and entity annotations. The most

[9] E.g., one may wish to compare neural networks with other machine learning methods (e.g., logistic regression, random forests) to evaluate the cost-benefit trade-off.

[10] A more detailed version of this table can be found in Appendix D Table 5

			Entities					Format		Annotations			Unk.
Dataset	**First Reported In**	**Versions**	**T**	**D**	**M**	**S**	**Md**	**PDF**	**LaTeX**	**HA**	**PwC**	**NLPP**	**Ann.**
ORKG-PwC	Kabongo et al. (2021)	v1-v7	✓	✓	✓	✗	✗	□	□	✗	✓	✗	□
NLP-TDMS	Hou et al. (2019)	v1-v3	✓	✓	✓	✓	✗	□	□	✗	✗	✓	□
PwC-LB	Kardas et al. (2020)	v1-v2	✓	✓	✓	✓	✗	□	□	✗	✓	✗	✗
SciREX	Jain et al. (2020)	-	✓	✓	✓	✗	✓	∼	∼	✓	✓	✗	✗
TDMS-Ctx	Kabongo et al. (2024)	v1-v6	✓	✓	✓	✓	✗	✓	✗	✓	✗	✓	✓
LEGOBench	Singh et al. (2024)	-	✓	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓
SciLead	Şahinüç et al. (2024)	-	✓	✓	✓	✓	✗	✓	✗	✓	✗	✗	✗

Table 2: Summary of datasets, detailing dataset variant (**V**), **Entities** captured (**T** = Task, **D** = Dataset, **M** = Metric, **S** = Score, **Md** = Method), **format** (PDF, LaTeX), **Annotations** (**HA** = Human Annotation, **PwC** = Papers with Code, **NLPP** = NLP Progress), inclusion of unknown annotations (**Unk. Ann.**) and the number of papers and tuples.

common format is ⟨*task, dataset, metric, score*⟩ (NLP-TDMS, (Hou et al., 2019), PwC-LB (Kardas et al., 2020), TDMS-Ctx (Kabongo et al., 2024), SciLead (Şahinüç et al., 2024)), while the most comprehensive format is ⟨*task, dataset, metric, score, method*⟩ (LEGOBench, (Singh et al., 2024)). These five datasets can be considered "complete" leaderboard datasets, as they include the score within the tuple.[11] In contrast, two related datasets do not include scores (ORKG-PwC (Kabongo et al., 2021), SciREX (Jain et al., 2020).[13]

3.2 Source of Annotations

Most datasets are assembled using manually curated leaderboards as a distant supervision source. For example, the first leaderboard dataset, *NLP-TDMS* (Hou et al., 2019), was derived from a community-maintained GitHub repository *NLP Progress*[14], tracking state-of-the-art NLP datasets and tasks. With the growing popularity of Paper with Code, many researchers turn to this resource to build ALG datasets, including ORKG-PwC, PwC-LB, SciREX, TDMS-Ctx and LEGOBench.

Not all datasets were created with manual annotations, however. Of the datasets derived from Papers with Code, only SciREX was subsequently corrected by a human annotator to ensure high accuracy. Similarly, for SciLead (Şahinüç et al., 2024), the leaderboard tuples ⟨task, dataset, metric, score⟩ were fully annotated by a single human annotator, prioritising quality but limiting dataset size due to the manual effort involved.

3.3 Format of the Papers

Datasets differ in publication formats. PDFs, though common, mix presentation with logical structure, whereas cleaner organisation. Some datasets use only one format—PDF (LEGOBench, SciLead) or LaTeX(TDMS-CtX)—while others provide both (NLP-TDMS, ORKG-PwC, PwC-LB). We note that this distinction is less important as tools like Grobid (Lopez, 2009) grow in maturity to transform PDF files into a logical structure format, such as XML.

4 Overview of ALG Evaluation Metrics

One key issue in the field has been the use of various metrics for ALG evaluation, hindering result comparisons. Appendix E lists all metrics used in leaderboard experiments. Below, we outline the key evaluation metrics used in prior work.

4.1 Precision, Recall and F1

Most work reports micro precision, recall, and F1, either for exact tuple matches or per entity class (e.g., task, metric). Some report macro variants, which offer deeper insights when frequent entities or tuples skew micro scores.

Although not explicitly stated, we believe that generally these scores are calculated per paper and then averaged. However, Singh et al. (2024) calculated precision and recall per leaderboard. Experimental results can vary significantly depending on whether metrics are averaged across papers, leaderboards, or entities/tuples. To demonstrate this significance, we replicated an experiment of Şahinüç et al. (2024) and found that if authors had used global averaging instead of per paper averages

[11] These datasets can sometimes be divided into further subsets based on the size of the leaderboard. E.g., the ORKG-PwC and NLP-TDMS datasets filter out leaderboards with less than five entries. Datasets can also be divided into pre-defined subsets. E.g., the ORKG datasets include pre-defined splits that correspond experimentation by Kabongo et al. (2024)[12].

[13] Although the paper does not mention recording the score, we found that the Github dataset includes a score. It is unclear whether this was added after the publication of the paper. https://github.com/allenai/SciREX

[14] https://github.com/sebastianruder/NLP-progress

the recall would differ by 12.61.[15] In Table 6 (Appendix E), we provide definitions of these metrics.

With the rise of generative AI with LLMs, there has been a need to explore string comparison metrics beyond exact match. For example, Kabongo et al. (2024) explored partial matches. We note that metrics are useful in open-domain settings, where multiple valid expressions may exist and exact matching is too restrictive.

4.2 Leaderboard Specific Metrics

In addition to standard retrieval metrics, Şahinüç et al. (2024) introduced four metrics for leaderboard evaluation: leaderboard recall (LR), paper coverage (PC), result coverage (RC), and average overlap (AO). LR measures the percentage of correctly identified test leaderboards. PC and RC compute the average percentage of correctly linked papers and scores per leaderboard, respectively. AO quantifies the overlap between generated and test leaderboards (Webber et al., 2010). These leaderboard-specific metrics go beyond entity- or tuple-level evaluation by directly measuring the quality of the reconstructed leaderboard as a whole. This shift is crucial: standard precision and recall metrics may overlook whether the extracted information actually supports leaderboard reconstruction, i.e. better reflect the end-goal of ALG systems. Hence, adopting such metrics is essential for driving progress in building end-to-end usable and trustworthy leaderboard extraction tools.

4.3 Granularity of Science Concepts

As science advances, scientific concepts evolve. For example, broad terms like neural LMs may split into finer categories (e.g. *pre-trained LMs* vs. *LLMs*), or sibling concepts may merge or become unevenly prominent (e.g. *abstractive summarisation* overtaking *extractive summarisation* with generative AI). Relatedly, capturing fine-grained method attributes, such as hyperparameters for neural networks, becomes increasingly important.

4.4 Extraction beyond Best Scores

Current ALG's focus on best scores limits its use to state-of-the-art comparisons and has drawn criticism for lacking real-world relevance. Ethayarajh and Jurafsky (2020) highlight that this emphasis neglects factors like fairness, compactness, and energy efficiency. Santy and Bhattacharya (2021) call for metrics beyond accuracy to better reflect practical utility. Braggaar et al. (2024) argue that rankings can mislead, as top models may underperform in practice. Rodriguez et al. (2021) emphasise that not all evaluation examples are equally informative, urging leaderboards to account for difficulty. Together, these critiques advocate for broader, more meaningful evaluation.

Including all experimental results introduces complexity, both methodologically (e.g. an LLM must extract more tuples, though many LLMs cannot output that many tokens) and from a user perspective (e.g. users must interpret a more complex leaderboard instead of a traditional one).

[15]The authors conducted a zero-shot experiment evaluated using exact match. They reported a recall of 47.53 when averaging per paper, whereas the recall would have been 34.92 if averaged globally across all tuples.

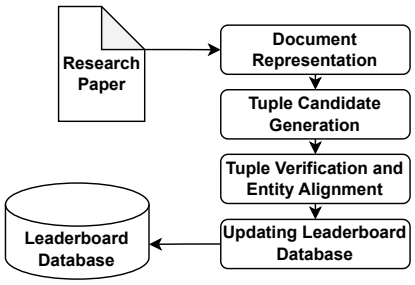

Figure 2: ALG Unified Conceptual Framework.

5 ALG Unified Conceptual Framework

To allow AI system builders to make system design choices based on research outcomes from ALG, we present the *ALG Unified Conceptual Framework*. For example, to build an ML leaderboard system, engineers may want to use the conceptualisation as inspiration for modules in a system architecture or agents in an Agentic AI system.

This conceptualisation is based on our analysis of the papers outlined in Table 1. Figure 2 illustrates these conceptual components and we provide examples of the methods for these components below, noting not all works include every component, reflecting differing research focuses.

The purpose of this conceptualisation is threefold: to (1) guide future researchers entering ALG research or building ALG systems; (2) organise the ALG experimentation space; and (3) understand the system-level importance of contributions.

5.1 Document Representation

We note that several papers focus on finding the best representation of paper contents, whether starting from PDF or structured formats like LaTeXõr

XML. Such representations help highlight key information, especially when later ML components must process limited input text.

For example, approaches using pre-trained language models (e.g. BERT), document representation is crucial due to input length limits (Hou et al., 2019). Hou et al. (2019) and Kabongo et al. (2021) used *document surrogates* like "DocTAET" (title, abstract, experimental setup, tables). Document representation can be more granular; for example, Jain et al. (2020) use entity chains to detect tuples.

Even with LLMs and their larger context windows, document representation remains important. Although LLMs can process full papers, the representation affects which information is used. Kabongo et al. (2024), for example, compare filtered document views with full-text inputs to assess effectiveness.

5.2 Tuple Candidate Generation

Given a document representation, this component extracts key contextual experimental attributes (e.g., task, dataset) and the result. There are various ways to extract this information, based on how domain knowledge is used.

5.2.1 Regarding Closed Domain Approaches

For PTE closed domain approaches, entities are generally defined in a finite set (PTE class). Any candidate tuples must be composed of these predefined entities and any new combination is acceptable. For example, systems can identify the key scientific concepts (e.g., extracting experiment attributes from relevant tables (Kardas et al., 2020)) to compose the tuples. For PTT approaches, the aim is to match the predefined tuple with the source document, in order to check for an improvement in performance. Hou et al. (2019) frame this as a Natural Language Inference (NLI) task, to see whether the tuple is inferred by the document representation.

5.2.2 Regarding Open Domain Approaches

For open-domain approaches, tuples may include entities beyond a predefined list. For example, in SciREX (Jain et al., 2020), an entity detector identifies spans corresponding to task, data set, metric, or method. These unbounded entities are then used to compose tuples. However, the authors do not specify how the extracted tuples would update the leaderboard database.

In Şahinüç et al. (2024), detected entities correspond to concepts that fall into two categories: (1) unseen (i.e., new) and (2) seen. Using a leaderboard database that is initially empty, entities are checked for corresponding entries, with either an exact match or a partial match. If a match exists, the existing form in the database is used as the canonical representation for that concept. This can be viewed as a data normalisation step. For all unmatched entities, these are treated as unseen, and a new database entry is created for it.

5.2.3 A Note on Score Extraction

Despite being central to ALG, only a handful of works (Hou et al., 2019; Kardas et al., 2020; Singh et al., 2024; Kabongo et al., 2024; Şahinüç et al., 2024) extract best scores. Other work focused on extracting the experimental conditions. We note that this is a precursor to finding the full tuple for ALG (identifying experimental conditions to which the best score belongs). For works that extract best scores, methods vary. Hou et al. (2019) apply heuristics based on orthographic features (boldface), whereas Kardas et al. (2020) use more complex inferences, classifying table cells as numeric or non-numeric. Extracted quantities are normalised and the extreme (maximum or minimum) score is kept based on the metric. Earlier models used dedicated methods to align scores with conditions, whereas recent LLM prompting extracts entire tuples, including scores, with a single task-based prompt (Kabongo et al., 2024; Singh et al., 2024; Şahinüç et al., 2024).

5.3 Tuple Verification and Entity Alignment

For each extracted tuple, the system should verify its correctness, especially for LLM-based approaches, which risk hallucinations. Pre-LLM methods often implicitly included this step within the extraction process. For example, by framing the tuple generation task as an NLI problem, Hou et al. (2019) extract tuples that are aligned with the source content *and* entailed by the source text, essentially performing verification. Others use partial alignment of the tuple at the entity level, such as using a Bayesian model to map different equivalent referring expressions to a canonical value (Kardas et al., 2020).

5.4 Updating Leaderboard Database

Once a tuple is verified, the final step is updating the leaderboard database. Kardas et al. (2020) link

experimental conditions to existing Papers with Code entries. Data may be normalised prior to this step (Şahinüç et al., 2024), and filtered to exclude, for example, ablation studies (Kardas et al., 2020). Most prior work does *not* detail this step, as the focus lies on NLP techniques for extraction rather than their downstream application, despite often being motivated by it.

6 ALG Benchmarking Guidelines

6.1 Open versus Closed Domain Reporting

We recommend that researchers report results for both open- and closed-domain scenarios. Closed-domain, which assumes predefined entities and tuples, provides the simplest case and typically yields the highest accuracy. Open-domain, by contrast, does not rely on predefined knowledge and thus represents the most challenging case. However, in practical applications, scenarios will typically fall between these extremes. To ensure that benchmarking captures this full range of difficulty, and to allow comparisons across studies, we advise that researchers always include results for both domains. Including both allows to assess the feasibility of leaderboard extraction under both the most constrained and the most unconstrained settings, which reflects the diversity of real-world conditions.

6.2 Dataset Reporting

We recommend that researchers report results on publicly available datasets as a minimum requirement. We highlight **SciLead** and **LEGOBench** as two suitable options. SciLead is valuable for its fully human-curated annotations, ensuring high quality. LEGOBench offers the largest dataset with broad tuple coverage, enabling large-scale benchmarking across diverse tasks and methods. These two datasets are complementary: SciLead provides a gold standard for high-accuracy evaluation, while LEGOBench allows robust assessment at scale. The feasibility of achieving broader and more informative evaluations strongly depends on ensuring open access to such datasets. Fortunately, SciLead and LEGOBench are fully open-source and thus support the practical feasibility of standardised evaluation without subscription or copyright barriers. However, a limitation of both datasets is that they only cover a restricted set of metadata attributes and focus solely on extracting the best results per paper. Therefore, in the next section (§7.6), we recommend that researchers develop more comprehensive datasets that include all reported results and richer metadata.

6.3 Metrics

Researchers should report precision, recall, and F1 as both **micro** and **macro** scores. Micro scores capture overall accuracy, favouring frequent entries, while macro scores weight papers, leaderboards, or entities equally and better reflect performance across varied result types. Reporting both provides balance, but most importantly researchers must clearly state the averaging method used (e.g. per paper, per leaderboard, or global).

In open-domain settings, exact string matching may be overly restrictive. We recommend reporting **partial match metrics**, which account for fuzzy or approximate matches. Such metrics better capture performance when multiple valid surface forms exist for the same scientific concept. This reflects real-world feasibility more accurately.

To assess practical usability for leaderboard construction, researchers should report **leaderboard-specific metrics**. In particular, we highlight leaderboard recall (LR), paper coverage (PC), result coverage (RC), and average overlap (AO). These metrics provide insights into how effectively extracted tuples populate leaderboards. Leaderboard recall reflects whether leaderboards are correctly identified. Paper coverage measures whether all relevant papers are linked. Result coverage assesses the proportion of extracted results, and average overlap quantifies agreement between generated and ground truth leaderboards.

When possible, results should also be analysed across **fine-grained scientific concepts**. For example, extraction accuracy should be reported not only at the tuple level, but also separately for tasks, datasets, metrics, methods, and scores. This supports a nuanced understanding of performance, especially where new or rarely seen concepts may be difficult to extract.

7 ALG Challenges and New Directions

To help guide ALG researchers and system designers to potentially novel capabilities, we list in this section challenges and new directions for ALG.

7.1 New or Unseen Entities

The 2024 shared task on ALG (D'Souza et al., 2024) highlights that many aspects of the task are still unsolved. It includes closed and open domain

subtasks, with the latter involving new entity detection.[16] Indeed, Kabongo et al. (2023a) showed that ML performance in extracting tuples with new entities (i.e., new scientific concepts, such as a newly introduced ML task or dataset) is much lower than extracting tuples with previously observed entities. In production, a challenge will be the feasibility of canonicalisation and disambiguation of these newly introduced ML entities. New entities often have ambiguous and inconsistent naming. For example, a newly introduced dataset might be referred to in short and long forms or with typos. In practice, feasibility depends on having automated canonicalisation methods that can cluster or align different surface forms of unseen entities. Without this, leaderboard entries will fragment into inconsistent records, undermining usability.

7.2 Document Representation

Representing source paper content remains an open challenge, even with LLMs' larger context windows. Kabongo et al. (2024) found that using the full document with DocTAET led to worse tuple extraction, underscoring the need for representations that balance coverage and minimise irrelevant content during inference. Another practical feasibility consideration is that LLMs with larger context windows are more expensive, making it desirable for users to adopt document representations that allow feasible use of smaller, more efficient models.

7.3 Extracting Numerical Scores

In most cases, the performance of tuple extraction, including scores, is significantly lower than that of tuples containing only the experimental conditions (which typically has F1 scores > 80), highlighting the difficulty of score extraction(Kardas et al., 2020; Hou et al., 2019; Yang et al., 2022; Şahinüç et al., 2024). For example, in recent work by Şahinüç et al. (2024), score extraction using GPT-4 achieved an F1 score of approximately 70.

Feasibility of extracting scores from a practical perspective goes further: not only must scores be extracted accurately, but extraction must be robust across various expressions of results. Systems must also handle ambiguous cases, such as ranges, averages, or multiple competing values. Current systems fall short in this respect, limiting the feasibility of fully automated leaderboard generation.

[16]The organisers refer to these as *few-shot* and *zero-shot*, referring on current ML terminology.

7.4 Feasibility of Extraction at Scale

Most research papers benchmark ALG systems on dozens or hundreds of papers. However, production-grade leaderboards such as Papers with Code integrate tens of thousands of papers. Extracting tuples at this scale introduces feasibility challenges in computational efficiency and LLM inference cost. Practical implementation of an always-updating leaderboard requires optimised batching, caching strategies, and asynchronous processing.

7.5 Generalisability beyond ML

A promising direction for future research is to explore the generalisability of ALG beyond ML. Domains like material science and biomedicine also report experimental results but use more varied formats and less standardised terminology. Key challenges include handling heterogeneous result expressions, complex domain language, and diverse contextual cues.

7.6 Comprehensive Leaderboards

A key direction for future research is the development of comprehensive leaderboards. By comprehensive, we mean not only *vertically*, by including all experimental results rather than only the best, but also *horizontally*, by capturing richer metadata (e.g., hyperparameters). A necessary first step is the creation of a novel dataset to benchmark both existing and new techniques.

8 Conclusion

In the position paper, we provide the **first overview of ALG research**, which reveals substantial diversity in problem framing and benchmarking practices. To address this fragmentation, we propose an **ALG unified conceptual framework** and present **ALG benchmarking guidelines**. Furthermore, our **first overview of ALG research to date** revealed that the scope of current leaderboards is limited. Therefore, one key recommendation in our list of **challenges and new directions for ALG** is to expand leaderboard coverage. Future leaderboards should report all results, including baselines, ablations, and method variations, and enrich tuples with broader metadata (e.g. hyperparameters) to create a more informative resource. In support of this initiative, a continually updated reading list is maintained in a GitHub repository[17].

[17]Anonymous while under review: https://github.com/anonymous391860/leaderboard-survey-anonymous

Limitations

A limitation of this paper is the scope, as we solely focus on the automatic generation of ML leaderboards. We note that other disciplines also report experimental outcomes, although the nature of the experimental procedures may differ. For example, Ghosh et al. (2024) explores finetuning LLMs for schema-based information extraction in material science. Another example is Wang et al. (2024), which introduced SciDaSynth, an interactive system using LLMs to extract and synthesise structured knowledge from the scientific literature in the form of tables.

Ethics

This research is subject to the governance by the ethics board of ANONYMOUS. We note that our proposal for AI research is to facilitate decision-making by users, as opposed to complete automation of tasks. We note that data mining activities for scientific literature should comply with the terms and conditions of the publishers disseminating published work, noting that scientific text mining is often consider to be fair use of copyright material. The use of AI for data mining in this case is on public domain material.

References

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

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, et al. 2023. The falcon series of open language models. *arXiv preprint*.

Anthropic. 2024. Zephyr Beta: An advanced LLM by anthropic. https://www.anthropic.com/.

arXiv. 2025. arXiv monthly submissions statistics. Accessed: 2025-01-20.

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

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

Anouck Braggaar, Linwei He, and Jan De Wit. 2024. Our dialogue system sucks—but luckily we are at the top of the leaderboard!: A discussion on current practices in NLP evaluation. In *Proceedings of the 6th ACM Conference on Conversational User Interfaces*, pages 1–5.

Davide Buscaldi, Anne-Kathrin Schumann, Behrang QasemiZadeh, Haïfa Zargayouna, and Thierry Charnois. 2018. SemEval-2018 task 7: Semantic relation extraction and classification in scientific papers. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 244–257, New Orleans, Louisiana. Springer, Association for Computational Linguistics.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* ChatGPT quality. Accessed: 2025-01-20.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *Advances in Neural Information Processing Systems*, 32.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jennifer D'Souza, Salomon Kabongo, Hamed Babaei Giglou, and Yue Zhang. 2024. Overview of the CLEF 2024 simpletext task 4: SOTA? tracking the state-of-the-art in scholarly publications. *Working Notes of CLEF*.

Markus Eberts and Adrian Ulges. 2020. Span-based joint entity and relation extraction with transformer pre-training. In *ECAI 2020*, pages 2006–2013. IOS Press.

Kawin Ethayarajh and Dan Jurafsky. 2020. Utility is in the eye of the user: A critique of NLP leaderboards. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4846–4853, Online. Association for Computational Linguistics.

Satanu Ghosh, Neal R Brodnik, Carolina Frey, Collin Holgate, Tresa M Pollock, Samantha Daly, and Samuel Carton. 2024. Toward reliable ad-hoc scientific information extraction: A case study on two materials datasets. *arXiv preprint:2406.05348v1*.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks*, 18(5-6):602–610.

Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly. 2019. Identification of tasks, datasets, evaluation metrics, and numeric scores for scientific leaderboards construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5203–5213, Florence, Italy. Association for Computational Linguistics.

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

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339.

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

Mohamad Yaser Jaradeh, Allard Oelen, Kheir Eddine Farfar, Manuel Prinz, Jennifer D'Souza, Gábor Kismihók, Markus Stocker, and Sören Auer. 2019. Open research knowledge graph: next generation infrastructure for semantic scholarly knowledge. In *Proceedings of the 10th international conference on knowledge capture*, pages 243–246.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint:2310.06825v1*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Salomon Kabongo, Jennifer D'Souza, and Sören Auer. 2023a. Zero-shot entailment of leaderboards for empirical AI research. volume 2023-June.

Salomon Kabongo, Jennifer D'Souza, and Sören Auer. 2024. Effective context selection in llm-based leaderboard generation: An empirical study. In *International Conference on Applications of Natural Language to Information Systems*, pages 150–160. Springer.

Salomon Kabongo, Jennifer D'Souza, and Sören Auer. 2021. Automated mining of leaderboards for empirical AI research. volume 13133 LNCS.

Salomon Kabongo, Jennifer D'Souza, and Sören Auer. 2023b. ORKG-leaderboards: a systematic workflow for mining leaderboards as a knowledge graph. *International Journal on Digital Libraries*.

Marcin Kardas, Piotr Czapla, Pontus Stenetorp, Sebastian Ruder, Sebastian Riedel, Ross Taylor, and Robert Stojnic. 2020. AxCell: Automatic extraction of results from machine learning papers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8580–8594, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Patrice Lopez. 2009. GROBID: Combining automatic bibliographic data recognition and term extraction for scholarship publications. In *Research and Advanced Technology for Digital Libraries: 13th European Conference, ECDL 2009, Corfu, Greece, September 27-October 2, 2009. Proceedings 13*, pages 473–474. Springer.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. pages 3219–3232.

John MacFarlane. 2006–. *Pandoc: A Universal Document Converter*. John MacFarlane.

George Michalopoulos, Michal Malyska, Nicola Sahar, Alexander Wong, and Helen Chen. 2022. ICDBigBird: A contextual embedding model for ICD code classification. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 330–336,

Dublin, Ireland. Association for Computational Linguistics.

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

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Pedro Rodriguez, Joe Barrow, Alexander Miserlis Hoyle, John P. Lalor, Robin Jia, and Jordan Boyd-Graber. 2021. Evaluation examples are not equally informative: How should that change NLP leaderboards? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4486–4503, Online. Association for Computational Linguistics.

Sebastin Santy and Prasanta Bhattacharya. 2021. A discussion on building practical NLP leaderboards: the case of machine translation. *arXiv preprint:2106.06292v1*.

Mayank Singh, Rajdeep Sarkar, Atharva Vyas, Pawan Goyal, Animesh Mukherjee, and Soumen Chakrabarti. 2019. Automated early leaderboard generation from comparative tables. In *Advances in Information Retrieval: 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14–18, 2019, Proceedings, Part I 41*, pages 244–257. Springer.

Shruti Singh, Shoaib Alam, Husain Malwat, and Mayank Singh. 2024. Legobench: Scientific leaderboard generation benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14598–14613.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *arXiv preprint:2211.09085v1*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint:2312.11805v1*.

LMSYS Team. 2023. Vicuna: An open-source chatbot trained by fine-tuning llama. https://vicuna.lmsys.org/.

Chris Tensmeyer, Vlad I Morariu, Brian Price, Scott Cohen, and Tony Martinez. 2019. Deep splitting and merging for table structure decomposition. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 114–121. IEEE.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint:2307.09288v1*.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Nicholas Walker, Sanghoon Lee, John Dagdelen, Kevin Cruse, Samuel Gleason, Alexander Dunn, Gerbrand Ceder, A Paul Alivisatos, Kristin A Persson, and Anubhav Jain. 2023. Extracting structured seed-mediated gold nanorod growth procedures from scientific text with LLMs. *Digital Discovery*, 2(6):1768–1782.

Xingbo Wang, Samantha L Huey, Rui Sheng, Saurabh Mehta, and Fei Wang. 2024. SciDaSynth: Interactive structured knowledge extraction and synthesis from scientific literature with large language model. *arXiv preprint:2404.13765v1*.

William Webber, Alistair Moffat, and Justin Zobel. 2010. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)*, 28(4):1–38.

Sean Yang, Chris Tensmeyer, and Curtis Wigington. 2022. TELIN: Table entity linker for extracting leaderboards from machine learning publications.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32:5753–5763.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297.

Fatih Şahinüç, Thi Thao Tran, Yuliya Grishina, Yufang Hou, Bowen Chen, and Iryna Gurevych. 2024. Efficient performance tracking: Leveraging large language models for automated construction of scientific leaderboards. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Miami, Florida, USA. Association for Computational Linguistics.

A Related Work Beyond ALG

Entity Recognition and Relation Extraction from Scientific Text Entity and relation extraction from scientific papers gained attention in 2017 with the SemEval-2017 ScienceIE task, which focused on identifying key elements like processes, tasks, and materials in publications (Augenstein et al., 2017). The SemEval-2018 Task 7 advanced this by classifying relationships such as "uses", "compares", and "improves" between scientific concepts (Buscaldi et al., 2018). Datasets like SciERC (Luan et al., 2018), TDMSci (Hou et al., 2021), and Dmdd (Pan et al., 2023) further support entity extraction research. The methods developed for scientific entity and relationship extraction can be leveraged to generate scientific leaderboards automatically.

Structured Scientific Information Extraction A scientific leaderboard compares methods, highlighting the best-performing one. It is a specific case of structured scientific information comparison and meta-analysis. Research has focused on extracting structured information without emphasizing leaderboards. For example, Ghosh et al. (2024) explored LLMs for schema-based information extraction in material science. Walker et al. (2023) improved extraction of experimental procedures using fine-tuned language models, while Wang et al. (2024) introduced SciDaSynth, an interactive system using LLMs to extract and synthesise structured knowledge from scientific literature.

B Problem Framing Details

Different methodologies for extracting leaderboard tuples rely on distinct document representations. The document representation defines which sections of a research paper are used before extracting leaderboard-related information. DocTAET contains text from a **Doc**ument's **T**itle, **A**bstract, **E**xperimental Setup, and **T**able information. DocREC consists of text from a **Doc**ument's **R**esults, **E**xperiments, and **C**onclusion sections. Some approaches extract content from the full paper, while others focus specifically on tables or citation tables. In Table 3, we show for each proposed methodology which document representation they use.

Methodology	**Document Representation**
TDMS-IE (Hou et al., 2019)	DocTAET*, SC
ORKG-TDM (Kabongo et al., 2021)	DocTAET
ORKG-LB (Kabongo et al., 2023b)	DocTAET
PI Graph (Singh et al., 2019)	Citation Tables
AXCELL (Kardas et al., 2020)	Full Paper & Tables
SciREX-IE (Jain et al., 2020)	Full Paper
TELIN (Yang et al., 2022)	Full Paper & Tables
TDMS-PR (Kabongo et al., 2024)	DocREC†
MS-PR (Singh et al., 2024)	Full Paper
TDMR-PR (Şahinüç et al., 2024)	Full Paper & Tables

* Hou et al. (2019) perform ablation studies with variations of DocTAET.
† Kabongo et al. (2024) compare the performance of three document representations: DocREC, DocTAET, and the Full Paper.

Table 3: Overview of the **Methodologies**. **Document Representation**: The content extracted from the paper before extracting the leaderboard tuples.

C Methodology Details

In this section, we provide a summary of all the proposed ALG methodologies, and in Table 4, we list for each methodology which language models it uses.

TDMS-IE Hou et al. (2019) propose TDMS-IE, a methodology to automatically extract ⟨task, dataset, metric, score⟩ tuples from research papers. The first step of TDMS-IE is extracting the document representation and the score context from the research paper. The document representation, DocTAET, covers the title, abstract, experimental setup, and table information. The title and abstract help predict the task, while the experimental setup and table information assist in identifying the dataset and metric. A second document-based structure, the score context, SC, represents contents from tables, since the work relies on table-based (and formatting, i.e., bold font) heuristics to generate candidate tuples. The SC captures the table caption and column headers corresponding to each bold-faced numeric score in each table of the research paper. This is used in conjunction with formatting-based heuristics to identify candidates for the best score of a ⟨task, dataset, metric⟩ tuple.[18] Hou et al. (2019) frame the problem as a natural language inference (NLI) task using two entailment models: 1) DocTAET-TDM and 2) SC-DM. Each model generates a tuple hypothesis (a Task-Dataset-Metric, or TDM, tuple for DocTAET-TDM; a Score-Dataset-

[18] For example, bold-faced scores are most likely to be best score.

Metric tuple for SC-DM), by searching for candidate argument combinations from a "taxonomy" (that is, a knowledge base) of previously observed tuples. A fine-tuned BERT model (for NLI) predicts whether a candidate tuple can be inferred from DocTAET, inferring links between the paper's text and the predefined canonical labels for the *Task, Dataset, and Metric*, as represented in the taxonomy. For instance, the model can recognise that "Rg-2" and "ROUGE-2" refer to the same metric. Similarly, the SC-DM infers entailment relationships between the SC document representations and dataset-metric tuples. Both models use the BERT model limited to 512 tokens (Devlin et al., 2019), although newer models with larger token capacities may improve performance.

PI Graph Singh et al. (2019) introduce the performance improvement graph (PI Graph) to rank research papers based on their performance. This graph is constructed from *performance tables*, which compare the methodologies and results of a paper with those from previous works. Citations within these tables create edges between papers, reflecting performance improvements. However, the authors do not detail how the performance tables are identified, extracted, or processed. The focus of this work is on ranking papers by performance, not on the extraction of leaderboard tuples, which falls outside the scope of their methodology.

AxCell Kardas et al. (2020) introduce AxCell, a pipeline for automatically extracting results from machine learning papers. AxCell first categorises tables into leaderboard, ablation, or irrelevant types using the ULMFiT classifier (Howard and Ruder, 2018). For leaderboard and ablation tables, each cell is classified as a dataset, metric, paper model, cited model, or other. BM25 (Robertson et al., 2009) is employed to extract relevant context from the paper for each cell. A generative model, based on the naive Bayes assumption, then links numeric cells to predefined leaderboards. Finally, the system filters out cited models, low-scoring links, and inferior results, retaining only the top results for each leaderboard.

SciREX-IE Jain et al. (2020) introduce SciREX-IE, a methodology for extracting N-ary relations from research papers. The process starts by extracting raw text and section information from documents (excluding figures, tables, and equations). SciREX-IE encodes the text in two steps: first, section-level token embeddings are obtained using SciBERT (Beltagy et al., 2019), followed by a BiLSTM (Graves and Schmidhuber, 2005) to capture cross-section dependencies. A BIOUL-based CRF tagger identifies and classifies mentions using BERT-BiLSTM embeddings, which are created by combining token embeddings with additional features. The system classifies mentions as salient or not and performs coreference resolution using the SciBERT embeddings, clustering mentions into entities. Salient clusters are then used for relation extraction, with document-level embeddings aggregating section data. The model jointly optimises mention identification, saliency classification, and relation extraction during training.

ORKG-TDM Kabongo et al. (2021) propose ORKG-TDM, a methodology to extract ⟨task, dataset, metric⟩ tuples from research papers. The authors refer to their approach as the ORKG-TDM, as it is integrated into a scholarly knowledge platform called Open Research Knowledge Graph (ORKG) (Jaradeh et al., 2019). ORKG-TDM follows a similar approach to TDMS-IE (Hou et al., 2019) by framing the tuple extraction problem as an entailment problem, but uses a single-step approach. As in TDMS-IE, DocTAET is the document representation, and leaderboard tuples coming from a predefined taxonomy are the hypotheses. New to ORKG-TDM is a task-specific parameter for the number of false triples per paper. While Hou et al. (2019) conducted experiments with only the original BERT model for TDMS-IE, Kabongo et al. 2021, in implementing the ORKG-TDM methodology, also experimented with the pre-trained SciBERT model (Beltagy et al., 2019), designed for scientific text, and XLNet (Yang et al., 2019), an autoregressive transformer capable of handling contexts longer than BERT's 512-token maximum.

TELIN Yang et al. (2022) proposed TELIN, a methodology to extract ⟨task, dataset, model, method⟩ tuples from research papers. TELIN begins by converting unstructured PDFs into structured documents, using YOLO to detect paragraphs, headings, captions, and tables (Redmon et al., 2016). SPLERGE is then applied to extract table components such as rows, columns, and cells (Tensmeyer et al., 2019). For NER, TELIN uses SpERT, a BERT-based model pre-trained on the SCiERC dataset, to classify scientific entities into categories like task, method, dataset, and evalua-

Methodology	Language Models
TDMS-IE (Hou et al., 2019)	BERT (Devlin et al., 2019)
ORKG-TDM (Kabongo et al., 2021)	XLNet (Yang et al., 2019), SciBERT (Beltagy et al., 2019), BERTbase (Devlin et al., 2019)
ORKG-LB (Kabongo et al., 2023b)	BERT (Devlin et al., 2019), SciBERT (Beltagy et al., 2019), XLNet (Yang et al., 2019), BigBird (Michalopoulos et al., 2022)
PI Graph (Singh et al., 2019)	Undefined
AxCell (Kardas et al., 2020)	ULMFiT classifier (Howard and Ruder, 2018), BM25 (Robertson et al., 2009)
SciREX-IE (Jain et al., 2020)	SciBERT (Beltagy et al., 2019), BiLSTM (Graves and Schmidhuber, 2005)
TELIN (Yang et al., 2022)	SpERT (Eberts and Ulges, 2020)
TDMS-PR (Kabongo et al., 2024)	Llama 2 (Touvron et al., 2023), Mistral (Jiang et al., 2023)
MS-PR (Singh et al., 2024)	Falcon (Almazrouei et al., 2023), Galactica (Taylor et al., 2022), Llama 2 (Touvron et al., 2023), Llama 3 (Dubey et al., 2024), Mistral (Jiang et al., 2023), Vicuna (Chiang et al., 2023), Zephyr (Tunstall et al., 2023), Gemini (Team et al., 2023), GPT-4 (Achiam et al., 2023)
TDMR-PR (Şahinüç et al., 2024)	Llama 2 (Touvron et al., 2023), Llama 3 (Dubey et al., 2024), Mixtral (Jiang et al., 2024), GPT-4 (Achiam et al., 2023)

Table 4: Overview of the language models used in each methodology, demonstrating how the methodologies have (logically) adopted more advanced models over time as discussed in Section 5.

tion metric (Eberts and Ulges, 2020). String matching between these entities and non-numeric table cells is performed using fuzzy search to handle non-exact matches and acronyms. Tuples are formed when at least three of the four entities (task, dataset, metric, model) are identified within the table and its caption. These extracted leaderboards are stored in a shared knowledge base, which is iteratively refined to discover more entities across documents. A human review stage prioritises uncertain entities, using feedback to fine-tune SpERT, iterating until entity prediction stabilises.

ORKG-LB Kabongo et al. (2023b) introduced ORKG Leaderboard (ORKG-LB), a follow-up methodology of ORKG-TDM (Kabongo et al., 2021). ORKG-LB focuses on the extraction of the ⟨task, dataset, metric⟩ tuples by framing the extraction task as an entailment problem. ORKG-LB starts by allowing users to input a LaTeX or PDF version of the research paper. ORKG-LB uses the GROBID parser (Lopez, 2009) for PDF files and PANDOC (MacFarlane, 2006–) to convert LaTeX files into XML TEI markup. Then, ORKG-LB extracts DocTAET (Hou et al., 2019), focusing on sections likely to contain task–dataset–metric mentions, reducing noise and enhancing generalisation. For training the inference, for each paper, positive and negative samples of tuples are required. For the number of false triples per paper, ORKG-LB relies on the same task-specific parameter as used for ORKG-TDM. For the inference model, the authors of ORKG-LB experiment with four different transformer model variants: BERT (Devlin et al., 2019), SciBERT (Beltagy et al., 2019), XLNet (Yang et al., 2019) and BigBird (Zaheer et al., 2020).

TDMS-PR The work of Kabongo et al. (2024) experiments with prompting LLMs to extract ⟨task, dataset, metric, score⟩ tuples from research papers, and we refer to this methodology as TDMS-PR. The authors experiment with different document representations provided to the LLM when prompting the LLM. They propose a novel document representation, DocREC, which comprises text from the results (R), experiments (E) and conclusions (C) sections. They compare the results when using DocREC to when using DocTAET (Hou et al., 2019) or DocFull, which is the full paper as document representation. On average, DocREC consists of more tokens than DocTAET, 1,586 versus 493, and by definition, DocFull is by default always the longest document representation. The authors experiment with LLMs from the Flan-T5 collection, Mistral 7B and Llama 3 7B.

MS-PR The authors of Singh et al. (2024) prompt an LLM to extract the ⟨method, score⟩ tuple given a research paper representation and a ⟨task, dataset, metric⟩ tuple; we refer to this as MS-PR. While both TDMS-PR (Kabongo et al., 2024) and MS-PR are prompt-based, their tuple scopes differ: TDMS-PR focuses on ⟨task, dataset, metric⟩, while MS-PR targets ⟨method, score⟩. Singh et al. (2024) experiment with MS-PR by using a wide range of LLMs: Falcon, Falcon Instruct, Galactica, Llama 2 (7B & 13B), Llama 2 Chat (7B & 13B), Mistral Instruct, Vicuna (7B & 13B), Zephyr Beta, Gemini Pro and GPT-4 (Almazrouei et al., 2023; Taylor et al., 2022; Touvron et al., 2023; Jiang et al., 2023; Team, 2023; Anthropic, 2024; Team et al., 2023; Achiam et al., 2023).

TDMR-PR The authors of Şahinüç et al. (2024) prompt an LLM to extract ⟨task, dataset, metric, score⟩ tuples, we refer to this method as TDMR-PR. First, TDMR-PR extracts the tuples from the papers via a retrieval-augmented generation method using an LLM. Second, depending on the domain (closed, hybrid, or, open), TDMR-PR normalises these tuples to a predefined taxonomy or creates new entries for novel tasks, datasets, or metrics. Lastly, TDMR-PR ranks the papers based on their performance, constructing or updating leaderboards accordingly.

D Dataset Details

Table 5 presents an extended version of Table 2, providing detailed information for each version of the included datasets. For every train, test, and validation split, we report the number of associated papers and extracted tuples. This table highlights the substantial diversity across datasets, which complicates direct comparisons between experiments.

E Definitions of Metrics

In this section, we define the micro and macro versions of the Precision, Recall, and F1 metrics for the ALG task. Based on our best guess, most of the existing works typically compute micro precision, micro recall, and micro F1 by first calculating these scores per paper and then averaging them. However, this is solely a best guess, and we know that, for example, Kabongo et al. (2024) and Singh et al. (2024) calculate the score on a leaderboard level. We recommend that future researchers either use these definitions of these metrics or explicitly specify if they average across a different dimension (e.g., across leaderboards), as the choice of the averaging method can significantly impact the final score.

$$\text{Micro P} = \frac{1}{P}\sum_{p=1}^{P}\frac{\sum_{i=1}^{N_p} TP_{p,i}}{\sum_{i=1}^{N_p}(TP_{p,i}+FP_{p,i})} \tag{1}$$

where P represents the total number of papers, and N_p represents the total number of extracted leaderboard tuples or entities, per paper p. The term $TP_{p,i}$ denotes the number of true positive instances for the i-th instance in paper p, while $FP_{p,i}$ represents the number of false positive instances for the i-th instance in the same paper. The precision is first computed for each individual paper before being averaged across all P papers.

Micro Recall measures the proportion of correctly identified leaderboard entities/tuples:

$$\text{Micro R} = \frac{1}{P}\sum_{p=1}^{P}\frac{\sum_{i=1}^{N_p} TP_{p,i}}{\sum_{i=1}^{N_p}(TP_{p,i}+FN_{p,i})} \tag{2}$$

where $FN_{p,i}$ represents the number of false negatives for the i-th instance in paper p.

Micro F1 is the *harmonic* mean of micro precision and micro recall, providing a balanced measure of extraction performance:

$$\text{Micro F1} = \frac{2 \times \text{Micro P} \times \text{Micro R}}{\text{Micro P} + \text{Micro R}} \tag{3}$$

We recommend also reporting the macro variants of these metrics to give more insight if some of the entries/tuples appear frequently and, therefore, disproportionally influence the micro scores. For *macro* metrics, we first average across all classes and then across P papers. Macro precision is given by:

$$\text{Macro P} = \frac{1}{P}\sum_{p=1}^{P}\frac{1}{C_p}\sum_{c=1}^{C_p}\frac{\sum_{i=1}^{N_{p,c}} TP_{p,c,i}}{\sum_{i=1}^{N_{p,c}}(TP_{p,c,i}+FP_{p,c,i})} \tag{4}$$

where C_p is the number of classes for each paper p.

Macro Recall is given by:

$$\text{Macro R} = \frac{1}{P}\sum_{p=1}^{P}\frac{1}{C}\sum_{c=1}^{C}\frac{\sum_{i=1}^{N_{p,c}} TP_{p,c,i}}{\sum_{i=1}^{N_{p,c}}(TP_{p,c,i}+FN_{p,c,i})} \tag{5}$$

And Macro F1 is given by:

$$\text{Macro F1} = \frac{1}{P}\sum_{p=1}^{P}\frac{1}{C}\sum_{c=1}^{C}\frac{2 \times \text{P}_{p,c} \times \text{R}_{p,c}}{\text{P}_{p,c} + \text{R}_{p,c}} \tag{6}$$

It is important to note that these definitions serve as an example of how micro and macro variations can be calculated when averaged at the paper level. However, these definitions can be easily adapted for calculations at the leaderboard level.

F An Overview of Experimental Results

We have compiled all the results we could find in the literature where researchers experiment with extracting leaderboard tuples and entities, evaluating these extractions using micro, partial micro, or macro precision, recall, and F1 scores. Table 7 presents an overview of these experiments. **This table highlights the complexity of comparing**

		Entities					Format		Annotations*			Unk.	Train Stats.		Test Stats.		Val. Stats.	
Paper	**V**	**T**	**D**	**M**	**S**	**Md**	**PDF**	**LaTeX**	**HA**	**PwC**	**NLPP**	**Ann.**	**#P**	**#T**	**#P**	**#T**	**#P**	**#T**
ORKG-PwC Dataset																		
Kabongo et al. (2021)	v1	✓	✓	✓	✗	✗	✓	✗	✗	✓	✗	✗	2,831†	11,724†	1,228†	5,060†	-	-
Kabongo et al. (2021)	v2	✓	✓	✓	✗	✗	✓	✗	✗	✓	✗	✓	3,753†	11,724†	1,608†	5,060†	-	-
Kabongo et al. (2023b)	v3	✓	✓	✓	✗	✗	✗	✓	✗	✓	✗	✗	587†	9,614†	270†	4,096†	-	-
Kabongo et al. (2023b)	v4	✓	✓	✓	✗	✗	✗	✓	✗	✓	✗	✓	2,946†	9,614†	1,262†	4,096†	-	-
Kabongo et al. (2023b)	v5	✓	✓	✓	✗	✗	✓	✗	✗	✓	✗	✗	587†	9,614†	270†	4,096†	-	-
Kabongo et al. (2023b)	v6	✓	✓	✓	✗	✗	✓	✗	✗	✓	✗	✓	2,946†	9,614†	1,262†	4,096†	-	-
Kabongo et al. (2023a)	v7#	✓	✓	✓	✗	✗	✓	✗	✗	✓	✗	✓	-	-	1,000	1,925	-	-
NLP-TDMS Dataset																		
Hou et al. (2019)	v1	✓	✓	✓	✓	✗	✓	✗	✗	✗	✓	✗	124	325	118	281	-	-
Hou et al. (2019)	v2	✓	✓	✓	✓	✗	✓	✗	✗	✗	✓	✓	170	325	162	281	-	-
Kardas et al. (2020)	v3	✓	✓	✓	✓	✗	✗	✓	✗	✗	✓	✓	≤170	≤325	≤162	≤281	-	-
PwC-LB Dataset																		
Kardas et al. (2020)	v1	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✗	‡	‡	516	2,802	‡	‡
Yang et al. (2022)	v2	✓	✓	✓	✓	✗	✓	✗	✗	✓	✗	✗	-	-	516	2,802	-	-
SciREX Dataset																		
Jain et al. (2020)		✓	✓	✓	✗	✓	∼	∼	✓	✓	✗	✗	≤438	∇	≤438	∇	≤438	∇
TDMS-Ctx Dataset																		
Kabongo et al. (2024)	v1§	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✓	11,807	402,409	1,326	33,863	-	-
Kabongo et al. (2024)	v2§	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✓	12,388	415,788	1,401	34,799	-	-
Kabongo et al. (2024)	v3§	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✓	10,058	415,788	1,105	31,213	-	-
Kabongo et al. (2024)	v4§	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✓	11,807	402,409	746	14,604	-	-
Kabongo et al. (2024)	v5§	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✓	12,388	415,788	789	14,800	-	-
Kabongo et al. (2024)	v6§	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✓	10,058	415,788	595	14,273	-	-
LEGOBench Dataset																		
Singh et al. (2024)		✓	✓	✓	✓	✓	✓	✗	✗	✓	✗	✓	-	-	◊	43,105	-	-
SciLead Dataset																		
Şahinüç et al. (2024)		✓	✓	✓	✓	✗	✓	✗	✓	✗	✗	✗	-	-	43	⊘	-	-

* For annotations, we distinguish between human annotations (HA), Papers with Code (PwC) and NLP Progress (NLPP), however PwC includes partial human annotation, and domain experts fully curated NLP Progress via GitHub pull requests. ∼ Use LaTeX if available; otherwise, default to PDF. ‡Different data for training (unlabelled arXiv papers and segmented tables) and validation (linked results). †Two-fold cross-validation: 70% train, 30% test, with averaged results. ◊ 9,847 leaderboards, and the number of papers are unspecified. § v1–v3 are few-shot experiment datasets with document representations: v1 (DocFULL), v2 (DocREC), and v3 (DocTAET). v4–v6 are zero-shot experiment datasets with the same representations: v4 (DocFULL), v5 (DocREC), and v6 (DocTAET). the same data source as v2, but with updated timestamps and no overlap with v2. ∇ An average of 5 tuple annotations per paper. ⊘ Unspecified, with 138 unique tuples reported.

Table 5: This table summarises the datasets from multiple research papers, detailing dataset variant (**V**), **Entities** captured (**T** = Task, **D** = Dataset, **M** = Metric, **S** = Score, **Md** = Method), **format** (PDF, LaTeX), **Annotations** (**HA** = Human Annotation, **PwC** = Papers with Code, **NLPP** = NLP Progress), and inclusion of unknown annotations (**Unk. Ann.**). Additionally, the table includes **Train**, **Test**, and validation (**Val.**) statistics (**Stats.**): the number of papers (**#P**) and tuples (**#T**).

different results due to the diversity of problem framing (e.g. closed versus open domain), datasets and metrics.. We omitted details on how the scores were averaged (e.g., across papers or leaderboards), as this information is often not reported in many studies. These differences in averaging methods also complicate direct comparisons between works. Please note that there may be additional subtle variations in the experimental setup that are not captured in the table, which could prevent a fair comparison.

	Micro			Macro			Part. Micro		
Paper	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**F1**	**Other Metrics**
Hou et al. (2019)	✓	✓	✓	✓	✓	✓	✗	✗	None
Kabongo et al. (2021)	✓	✓	✓	✓	✓	✓	✗	✗	None
Kabongo et al. (2023b)	✓	✓	✓	✓	✓	✓	✗	✗	None
Kabongo et al. (2023a)	✓	✓	✓	✓	✓	✓	✗	✗	None
Kardas et al. (2020)	✓	✓	✓	✓	✓	✓	✗	✗	None
Jain et al. (2020)	✓	✓	✓	✗	✗	✗	✗	✗	None
Yang et al. (2022)	✓	✓	✓	✓	✓	✓	✗	✗	None
Kabongo et al. (2024)	✓	✗	✓	✗	✗	✗	✓	✓	None
Singh et al. (2024)	✓	✓	✗	✗	✗	✗	✗	✗	None
Şahinüç et al. (2024)	✓	✓	✓	✗	✗	✗	✗	✗	leaderboard recall (LR), paper coverage (PC), result coverage (RC), and average overlap (AO)

Table 6: Overview of evaluation metrics used in each paper.

	Micro			Macro			Part. Micro				
Reported In	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**F1**	**Dataset**	**Method**	**Experimental Setup**
Results of Extracting ⟨Task, Dataset, Metric⟩ for Closed Domain Problem Framing											
Hou et al. (2019)	60.2	73.1	66.0	54.1	65.9	56.6			NLP-TDMS-v1	TDMS-IE	
Hou et al. (2019)	29.4	42.0	34.6	24.9	43.6	28.1			NLP-TDMS-v1	EL†	
Hou et al. (2019)	56.8	23.8	33.6	56.8	30.9	37.3			NLP-TDMS-v1	MLC†	
Hou et al. (2019)	16.8	7.8	10.6	8.1	6.4	6.9			NLP-TDMS-v1	SM†	
Hou et al. (2019)	60.8	76.8	67.8	62.5	75.2	65.3			NLP-TDMS-v2	TDMS-IE	
Hou et al. (2019)	24.3	36.3	29.1	18.1	31.8	20.5			NLP-TDMS-v2	EL†	
Hou et al. (2019)	42.0	20.9	27.9	42.0	23.1	27.8			NLP-TDMS-v2	MLC†	
Hou et al. (2019)	36.0	19.6	25.4	31.8	30.6	31.0			NLP-TDMS-v2	SM†	
Hou et al. (2019)	68.6	40.3	50.8	29.6	29.1	28.1			NLP-TDMS-v2	TDMS-IE	TAE#
Hou et al. (2019)	50.0	23.7	32.2	20.8	20.1	19.4			NLP-TDMS-v2	TDMS-IE	TAT#
Hou et al. (2019)	47.9	14.2	21.9	11.3	11.3	10.7			NLP-TDMS-v2	TDMS-IE	TA#
Kardas et al. (2020)	65.8	58.5	61.9	56.0	55.8	54.1			NLP-TDMS-v3	AxCell	
Kardas et al. (2020)	53.4	66.3	59.2	57.1	66.1	58.5			NLP-TDMS-v3	TDMS-IE	
Kardas et al. (2020)	67.8	47.8	56.1	47.9	46.4	43.5			PwC-LB-v1	AxCell	
Kabongo et al. (2021)	76.4	66.4	71.1	63.5	64.1	61.4			NLP-TDMS-v1	ORKG-TDM	XLNet
Kabongo et al. (2021)	65.3	73.1	69.0	57.6	68.7	60.1			NLP-TDMS-v1	ORKG-TDM	SciBERT
Kabongo et al. (2021)	79.5	57.6	66.8	59.0	55.4	54.7			NLP-TDMS-v1	ORKG-TDM	BERT
Kabongo et al. (2021)	77.1	70.9	73.9	71.7	73.9	70.6			NLP-TDMS-v2	ORKG-TDM	XLNet
Kabongo et al. (2021)	79.6	63.3	70.5	68.1	67.5	65.5			NLP-TDMS-v2	ORKG-TDM	BERT
Kabongo et al. (2021)	65.7	76.8	70.8	65.7	77.2	68.3			NLP-TDMS-v2	ORKG-TDM	SciBERT
Kabongo et al. (2021)	95.1	92	93.5	92.3	93.5	91.7			ORKG-PwC-v1	ORKG-TDM	XLNet TAET#
Kabongo et al. (2021)	93.5	93.2	93.3	90.5	94.4	91.2			ORKG-PwC-v1	ORKG-TDM	XLNet TAT#
Kabongo et al. (2021)	95.0	90.5	92.7	91.6	93.1	91.2			ORKG-PwC-v1	ORKG-TDM	XLNet#
Kabongo et al. (2021)	95.7	88.3	91.8	91.7	92.1	90.8			ORKG-PwC-v1	ORKG-TDM	BERT
Kabongo et al. (2021)	94.2	89	91.5	89.2	91.5	89.2			ORKG-PwC-v1	ORKG-TDM	XLNet TAE#
Kabongo et al. (2021)	94.4	87.6	90.9	89.7	91.4	89.4			ORKG-PwC-v1	ORKG-TDM	SciBERT
Kabongo et al. (2021)	92.6	90	91.3	88.6	92.9	89.4			ORKG-PwC-v1	ORKG-TDM	XLNet TA#
Kabongo et al. (2021)	94.9	91.2	93.0	92.8	94.8	92.8			ORKG-PwC-v2	ORKG-TDM	XLNet
Kabongo et al. (2021)	95.5	89.1	92.1	92.8	93.9	92.4			ORKG-PwC-v2	ORKG-TDM	BERT
Kabongo et al. (2021)	94.1	88.5	91.2	90.9	93.4	91.1			ORKG-PwC-v2	ORKG-TDM	SciBERT
Kabongo et al. (2023b)	95.2	92.2	93.6	91.5	93.3	91.3			ORKG-PwC-v5	ORKG-LB	BigBERT
Kabongo et al. (2023b)	94.8	93.9	94.3	91.3	94.4	91.8			ORKG-PwC-v5	ORKG-LB	BERT
Kabongo et al. (2023b)	94.8	93.9	94.3	91.3	94.4	91.8			ORKG-PwC-v5	ORKG-LB	SciBERT
Kabongo et al. (2023b)	95.4	93.9	94.7	93.2	95.7	93.5			ORKG-PwC-v6	ORKG-LB	BERT
Kabongo et al. (2023b)	95.4	91.1	93.2	92.6	94.3	92.2			ORKG-PwC-v6	ORKG-LB	SciBERT
Kabongo et al. (2023b)	93.2	94.9	93.0	95.7	92.4	94.0			ORKG-PwC-v6	ORKG-LB	BigBERT
Kabongo et al. (2023b)	95.1	94.6	94.8	93.1	96.4	93.7			ORKG-PwC-v6	ORKG-LB	XLNet
Kabongo et al. (2023b)	95.4	88.0	91.5	91.2	92.3	90.6			ORKG-PwC-v3	ORKG-LB	BERT
Kabongo et al. (2023b)	93.7	86.0	89.7	89.4	91.7	89.2			ORKG-PwC-v3	ORKG-LB	SciBERT
Kabongo et al. (2023b)	93.6	85.3	89.3	87.5	88.7	86.6			ORKG-PwC-v3	ORKG-LB	BigBird
Kabongo et al. (2023b)	94.9	91.2	93.0	91.9	94.4	92.0			ORKG-PwC-v4	ORKG-LB	XLNet
Kabongo et al. (2023b)	96.0	90.0	92.9	93.5	94.2	92.8			ORKG-PwC-v4	ORKG-LB	BERT
Kabongo et al. (2023b)	94.6	88.6	91.5	91.7	93.9	91.6			ORKG-PwC-v4	ORKG-LB	SciBERT
Kabongo et al. (2023b)	94.6	87.2	90.7	90.7	91.6	89.7			ORKG-PwC-v4	ORKG-LB	BigBird
Kabongo et al. (2023a)	9.2	78.1	16.5	14.3	86.6	21.9			ORKG-PwC-v7*	ORKG-TDM	XLNet
Kabongo et al. (2023a)	14.1	72.9	23.6	20.1	83.4	28.9			ORKG-PwC-v7*	ORKG-TDM	BERT
Kabongo et al. (2023a)	10.4	81.7	18.4	16.2	89	24.4			ORKG-PwC-v7*	ORKG-TDM	BERT
Kabongo et al. (2023a)	10.1	76.8	17.8	14.9	86.4	22.7			ORKG-PwC-v7*	ORKG-TDM	XLNet
Şahinüç et al. (2024)	55.1	25.8	35.1						SciLead	AxCell	
Şahinüç et al. (2024)	40.7	39.5	40.1						SciLead	TDMR-PR	Llama 2+CS
Şahinüç et al. (2024)	35.9	34.9	35.4						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	58.4	52.1	55.1						SciLead	TDMR-PR	Mixtral+CS
Şahinüç et al. (2024)	55.7	48.8	51.0						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	62.0	58.1	60.0						SciLead	TDMR-PR	Llama 3+CS
Şahinüç et al. (2024)	77.1	72.6	74.8						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	69.0	63.8	66.3						SciLead	TDMR-PR	GPT-4+CS
Şahinüç et al. (2024)	75.3	70.4	72.8						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Task, Dataset, Metric⟩ for Open Domain Problem Framing											
Yang et al. (2022)	68.2	45.3	56.5	49.7	43.1	42.5			PwC-LB-v2	TELIN	
Results of Extracting ⟨Task, Dataset, Metric⟩ for Hybrid Domain Problem Framing											

Continued on next page.

	Micro			Macro			Part. Micro				
Reported In	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**F1**	**Dataset**	**Method**	**Experimental Setup**
Şahinüç et al. (2024)	27.23	22.99	24.93						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	27.89	24.48	26.07						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	50.75	45.30	47.87						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	56.08	51.89	53.90						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Task, Dataset, Metric, Score⟩ for Closed Domain Problem Framing											
Hou et al. (2019)	10.8	13.1	11.8	9.3	11.8	9.9			NLP-TDMS-v1	TDMS-IE	
Hou et al. (2019)	3.8	1.8	2.4	1.3	1.0	1.1			NLP-TDMS-v1	SM†	
Hou et al. (2019)	6.8	2.9	4.0	6.8	6.1	6.2			NLP-TDMS-v1	MLC†	
Kardas et al. (2020)	27.4	24.4	25.8	20.2	20.6	19.7			NLP-TDMS-v3	AxCell	
Kardas et al. (2020)	6.8	8.4	7.5	8.6	9.5	8.8			NLP-TDMS-v3	TDMS-IE	
Kardas et al. (2020)	37.4	23.2	28.7	24.0	21.8	21.1			PwC-LB-v1	AxCell	
Şahinüç et al. (2024)	32.59	13.67	19.26						SciLead	AxCell	
Şahinüç et al. (2024)	10.06	21.59	13.73						SciLead	TDMR-PR	Llama 2+CS
Şahinüç et al. (2024)	9.63	15.25	11.81						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	26.54	24.61	25.54						SciLead	TDMR-PR	Mixtral+CS
Şahinüç et al. (2024)	24.66	21.73	23.10						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	23.22	29.54	26.00						SciLead	TDMR-PR	Llama 3+CS
Şahinüç et al. (2024)	27.11	35.60	30.78						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	49.82	48.71	49.26						SciLead	TDMR-PR	GPT-4+CS
Şahinüç et al. (2024)	56.02	54.53	55.27						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Task, Dataset, Metric, Score⟩ for Open Domain Problem Framing											
Yang et al. (2022)	38.3	20.8	26.3	26.6	19.2	21.3			PwC-LB-v2	TELIN	
Results of Extracting ⟨Task, Dataset, Metric, Score⟩ for Hybrid Domain Problem Framing											
Şahinüç et al. (2024)	4.17	9.89	5.87						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	14.65	12.27	13.35						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	15.70	18.75	17.09						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	40.60	39.56	40.07						SciLead	TDMR-PR	GPT-4
Şahinüç et al. (2024)	51.01	51.03	51.02						SciLead	TDMR-PR	GPT-4 FS
Results of Extracting ⟨Task, Dataset, Metric, Method⟩ for Closed Domain Problem Framing											
Jain et al. (2020)	0.48	0.89	0.62						SciREX	TDMS-IE	
Results of Extracting ⟨Task, Dataset, Metric, Method⟩ for Open Domain Problem Framing											
Jain et al. (2020)	0.53	0.72	0.61						SciREX	SciREX-IE	
Results of Extracting ⟨Task⟩ for Closed Domain Problem Framing											
Kardas et al. (2020)	70.6	57.3	63.3	60.7	62.6	59.7			PwC-LB-v1	AxCell	
Kabongo et al. (2021)	97.4	93.6	95.5	93.7	94.8	93.6			ORKG-PwC-v1	ORKG-TDM	XLNet
Kabongo et al. (2023b)	96.8	95.9	96.4	94.3	97.2	95.0			ORKG-PwC-v6	ORKG-LB	XLNet
Kabongo et al. (2023b)	96.8	95.9	96.4	94.3	97.2	95.0			ORKG-PwC-v4	ORKG-LB	XLNet
Şahinüç et al. (2024)	68.98	58.52	63.32						SciLead	AxCell	
Şahinüç et al. (2024)	59.83	67.20	63.30						SciLead	TDMR-PR	Llama 2+CS
Şahinüç et al. (2024)	55.45	60.74	57.97						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	86.27	91.99	89.04						SciLead	TDMR-PR	Mixtral+CS
Şahinüç et al. (2024)	86.85	89.74	88.27						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	85.69	90.85	88.19						SciLead	TDMR-PR	Llama 3+CS
Şahinüç et al. (2024)	87.33	92.17	89.68						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	90.70	90.77	90.73						SciLead	TDMR-PR	GPT-4+CS
Şahinüç et al. (2024)	90.62	91.10	90.86						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Task⟩ for Open Domain Problem Framing											
Yang et al. (2022)	70.3	53.7	59.2	60.5	57.3	57.1			PwC-LB-v2	TELIN	
Kabongo et al. (2024)	31.89		13.97				54.92	24.05	TDMS-Ctx-v5	TDMS-PR	Llama2 7B ZS REC#
Kabongo et al. (2024)	24.56		21.75				43.46	38.48	TDMS-Ctx-v6	TDMS-PR	Llama2 7B ZS TAET#
Kabongo et al. (2024)	2.06		2.06				52.54	3.36	TDMS-Ctx-v4	TDMS-PR	Llama2 7B ZS Full#
Kabongo et al. (2024)	17.99		17.99				59.25	29.88	TDMS-Ctx-v5	TDMS-PR	Mistral 7B ZS REC#
Kabongo et al. (2024)	26.99		26.99				64.00	44.90	TDMS-Ctx-v6	TDMS-PR	Mistral 7B ZS TAET#
Kabongo et al. (2024)	0.22		0.56				62.50	0.56	TDMS-Ctx-v4	TDMS-PR	Mistral 7B ZS Full#
Kabongo et al. (2024)	34.10		20.93				51.13	31.37	TDMS-Ctx-v2	TDMS-PR	Llama2 7B FS REC#
Kabongo et al. (2024)	30.61		29.53				44.96	43.37	TDMS-Ctx-v3	TDMS-PR	Llama2 7B FS TAET#
Kabongo et al. (2024)	34.69		1.59				50.00	2.29	TDMS-Ctx-v1	TDMS-PR	Llama2 7B FS Full#

Continued on next page.

	Micro			Macro			Part. Micro				
Reported In	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**F1**	**Dataset**	**Method**	**Experimental Setup**
Kabongo et al. (2024)	37.65		26.77				55.90	39.75	TDMS-Ctx-v2	TDMS-PR	Mistral 7B FS REC[#]
Kabongo et al. (2024)	39.48		33.38				54.82	46.35	TDMS-Ctx-v3	TDMS-PR	Mistral 7B FS TAET[#]
Kabongo et al. (2024)	32.43		0.81				71.43	1.19	TDMS-Ctx-v1	TDMS-PR	Mistral 7B FS Full[#]
Results of Extracting ⟨Task⟩ for Hybrid Domain Problem Framing											
Şahinüç et al. (2024)	39.70	42.98	41.27						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	50.23	60.72	54.98						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	65.72	80.39	72.32						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	63.82	78.30	70.32						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Dataset⟩ for Closed Domain Problem Framing											
Kardas et al. (2020)	70.2	48.4	57.3	53.5	52.7	49.9			PwC-LB-v1	AxCell	
Kabongo et al. (2021)	96.6	91.5	94.0	92.9	93.6	92.4			ORKG-PwC-v1	ORKG-TDM	XLNet
Kabongo et al. (2023b)	96.2	95.4	95.8	93.8	96.7	94.4			ORKG-PwC-v6	ORKG-LB	XLNet
Kabongo et al. (2023b)	96.2	95.4	95.8	93.8	96.7	94.4			ORKG-PwC-v4	ORKG-LB	XLNet
Şahinüç et al. (2024)	63.66	33.87	44.22						SciLead	AxCell	
Şahinüç et al. (2024)	68.93	58.81	63.47						SciLead	TDMR-PR	Llama 2+CS
Şahinüç et al. (2024)	62.60	55.03	58.57						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	85.03	73.20	78.67						SciLead	TDMR-PR	Mixtral+CS
Şahinüç et al. (2024)	81.68	71.26	76.12						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	82.43	78.62	80.48						SciLead	TDMR-PR	Llama 3+CS
Şahinüç et al. (2024)	92.09	87.75	89.87						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	86.36	79.93	83.02						SciLead	TDMR-PR	GPT-4+CS
Şahinüç et al. (2024)	92.64	86.05	89.22						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Dataset⟩ for Open Domain Problem Framing											
Kabongo et al. (2024)	15.77		6.83				38.32	16.6	TDMS-Ctx-v5	TDMS-PR	Llama2 7B ZS REC[#]
Kabongo et al. (2024)	12.72		11.26				26.09	23.1	TDMS-Ctx-v6	TDMS-PR	Llama2 7B ZS TAET[#]
Kabongo et al. (2024)	20.34		1.30				38.98	2.49	TDMS-Ctx-v4	TDMS-PR	Llama2 7B ZS Full[#]
Kabongo et al. (2024)	23.40		11.80				41.73	21.05	TDMS-Ctx-v5	TDMS-PR	Mistral 7B ZS REC[#]
Kabongo et al. (2024)	20.41		14.32				38.89	27.29	TDMS-Ctx-v6	TDMS-PR	Mistral 7B ZS TAET[#]
Kabongo et al. (2024)	37.50		0.33				75.00	0.67	TDMS-Ctx-v4	TDMS-PR	Mistral 7B ZS Full[#]
Kabongo et al. (2024)	21.27		13.06				36.66	22.50	TDMS-Ctx-v2	TDMS-PR	Llama2 7B FS REC[#]
Kabongo et al. (2024)	17.29		16.68				31.48	30.36	TDMS-Ctx-v3	TDMS-PR	Llama2 7B FS TAET[#]
Kabongo et al. (2024)	29.59		1.36				39.80	1.82	TDMS-Ctx-v1	TDMS-PR	Llama2 7B FS Full[#]
Kabongo et al. (2024)	22.15		15.68				38.52	27.28	TDMS-Ctx-v2	TDMS-PR	Mistral 7B FS REC[#]
Kabongo et al. (2024)	21.89		18.51				38.73	32.75	TDMS-Ctx-v3	TDMS-PR	Mistral 7B FS TAET[#]
Kabongo et al. (2024)	32.43		0.57				48.65	0.85	TDMS-Ctx-v1	TDMS-PR	Mistral 7B FS Full[#]
Yang et al. (2022)	70.9	52.8	59.3	54.7	55.2	53.9			PwC-LB-v2	TELIN	
Results of Extracting ⟨Dataset⟩ for Hybrid Domain Problem Framing											
Şahinüç et al. (2024)	41.05	33.14	36.67						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	49.67	44.45	46.92						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	66.81	62.86	64.77						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	83.29	79.52	81.36						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Metric⟩ for Closed Domain Problem Framing											
Kardas et al. (2020)	68.8	58.5	63.3	58.4	60.4	56.5			PwC-LB-v1	AxCell	
Kabongo et al. (2021)	96.0	92.5	94.2	92.5	94.2	92.5			ORKG-PwC-v1	ORKG-TDM	XLNet
Kabongo et al. (2023b)	96.0	95.3	95.6	93.7	96.9	94.4			ORKG-PwC-v6	ORKG-LB	XLNet
Kabongo et al. (2023b)	96.0	95.3	95.6	93.7	96.9	94.4			ORKG-PwC-v4	ORKG-LB	XLNet
Kabongo et al. (2024)	26.77		11.72				41.73	18.28	TDMS-Ctx-v5	TDMS-PR	Llama2 7B ZS REC[#]
Kabongo et al. (2024)	19.19		16.99				30.60	27.09	TDMS-Ctx-v6	TDMS-PR	Llama2 7B ZS TAET[#]
Kabongo et al. (2024)	23.73		1.52				38.98	2.49	TDMS-Ctx-v4	TDMS-PR	Llama2 7B ZS Full[#]
Kabongo et al. (2024)	31.02		15.55				46.20	23.16	TDMS-Ctx-v5	TDMS-PR	Mistral 7B ZS REC[#]
Kabongo et al. (2024)	31.41		22.04				45.94	32.23	TDMS-Ctx-v6	TDMS-PR	Mistral 7B ZS TAET[#]
Kabongo et al. (2024)	37.50		0.33				87.50	0.78	TDMS-Ctx-v4	TDMS-PR	Mistral 7B ZS Full[#]
Kabongo et al. (2024)	22.74		13.96				35.82	21.99	TDMS-Ctx-v2	TDMS-PR	Llama2 7B FS REC[#]
Kabongo et al. (2024)	20.78		20.02				31.66	30.51	TDMS-Ctx-v3	TDMS-PR	Llama2 7B FS TAET[#]
Kabongo et al. (2024)	20.41		0.94				36.73	1.68	TDMS-Ctx-v1	TDMS-PR	Llama2 7B FS Full[#]
Kabongo et al. (2024)	26.38		18.70				40.18	28.49	TDMS-Ctx-v2	TDMS-PR	Mistral 7B FS REC[#]
Kabongo et al. (2024)	28.66		24.23				40.41	34.16	TDMS-Ctx-v3	TDMS-PR	Mistral 7B FS TAET[#]
Kabongo et al. (2024)	32.43		0.57				45.95	0.81	TDMS-Ctx-v1	TDMS-PR	Mistral 7B FS Full[#]

Continued on next page.

	Micro			Macro			Part. Micro				
Reported In	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**F1**	**Dataset**	**Method**	**Experimental Setup**
Şahinüç et al. (2024)	69.35	51.36	59.01						SciLead	AxCell	
Şahinüç et al. (2024)	67.36	61.41	64.25						SciLead	TDMR-PR	Llama 2+CS
Şahinüç et al. (2024)	71.51	65.49	68.37						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	76.56	71.78	74.09						SciLead	TDMR-PR	Mixtral+CS
Şahinüç et al. (2024)	76.72	67.20	71.65						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	87.02	81.41	84.12						SciLead	TDMR-PR	Llama 3+CS
Şahinüç et al. (2024)	94.90	89.48	92.11						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	86.36	81.49	83.85						SciLead	TDMR-PR	GPT-4+CS
Şahinüç et al. (2024)	88.18	86.46	87.31						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Metric⟩ for Open Domain Problem Framing											
Yang et al. (2022)	63.2	57.9	60.2	56.3	55.1	55.4			PwC-LB-v2	TELIN	
Results of Extracting ⟨Metric⟩ for Hybrid Domain Problem Framing											
Şahinüç et al. (2024)	61.24	59.34	60.28						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	78.72	71.19	74.77						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	94.90	88.90	91.80						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	92.21	89.27	90.72						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Score⟩ for Closed Domain Problem Framing											
Şahinüç et al. (2024)	45.32	18.41	26.18						SciLead	AxCell	
Şahinüç et al. (2024)	23.75	31.61	27.12						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	44.62	41.75	43.13						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	39.50	49.56	43.96						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	70.34	68.22	69.26						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Score⟩ for Open Domain Problem Framing											
Kabongo et al. (2024)	6.06		2.61				7.27	3.10	TDMS-Ctx-v5	TDMS-PR	Llama2 7B ZS REC[#]
Kabongo et al. (2024)	0.87		0.77				1.09	0.96	TDMS-Ctx-v6	TDMS-PR	Llama2 7B ZS TAET[#]
Kabongo et al. (2024)	5.08		0.33				8.47	0.54	TDMS-Ctx-v4	TDMS-PR	Llama2 7B ZS Full[#]
Kabongo et al. (2024)	9.98		5.04				11.46	5.75	TDMS-Ctx-v5	TDMS-PR	Mistral 7B ZS REC[#]
Kabongo et al. (2024)	1.71		1.20				2.03	1.41	TDMS-Ctx-v6	TDMS-PR	Mistral 7B ZS TAET[#]
Kabongo et al. (2024)	14.00		0.76				21.62	0.87	TDMS-Ctx-v4	TDMS-PR	Mistral 7B ZS Full[#]
Kabongo et al. (2024)	4.99		3.04				5.59	3.46	TDMS-Ctx-v2	TDMS-PR	Llama2 7B FS REC[#]
Kabongo et al. (2024)	1.18		1.14				1.43	1.38	TDMS-Ctx-v3	TDMS-PR	Llama2 7B FS TAET[#]
Kabongo et al. (2024)	5.10		0.23				8.16	0.37	TDMS-Ctx-v1	TDMS-PR	Llama2 7B FS Full[#]
Kabongo et al. (2024)	8.94		6.36				9.95	7.08	TDMS-Ctx-v2	TDMS-PR	Mistral 7B FS REC[#]
Kabongo et al. (2024)	2.21		1.87				2.65	2.25	TDMS-Ctx-v3	TDMS-PR	Mistral 7B FS TAET[#]
Kabongo et al. (2024)	9.6		0.56				14.52	0.84	TDMS-Ctx-v1	TDMS-PR	Mistral 7B FS Full[#]
Singh et al. (2024)	2.13								LEGOBench	MS-PR[‡]	Mistral Instr. 7B
Singh et al. (2024)	1.81								LEGOBench	MS-PR[‡]	Zephyr Beta 7B
Singh et al. (2024)	13.87								LEGOBench	MS-PR[‡]	Gemini Pro
Singh et al. (2024)	13.06								LEGOBench	MS-PR[‡]	GPT-4
Results of Extracting ⟨Score⟩ for Hybrid Domain Problem Framing											
Şahinüç et al. (2024)	23.75	31.61	27.12						SciLead	TDMR-PR	Llama 2
Şahinüç et al. (2024)	44.62	41.75	43.13						SciLead	TDMR-PR	Mixtral
Şahinüç et al. (2024)	39.50	49.56	43.96						SciLead	TDMR-PR	Llama 3
Şahinüç et al. (2024)	70.34	68.22	69.26						SciLead	TDMR-PR	GPT-4
Results of Extracting ⟨Method⟩ for Open Domain Problem Framing											
Singh et al. (2024)		0.010							LEGOBench	MS-PR[‡]	Falcon 7B
Singh et al. (2024)		0.002							LEGOBench	MS-PR[‡]	Falcon Instr. 7B
Singh et al. (2024)		0.000							LEGOBench	MS-PR[‡]	Galactica 7B
Singh et al. (2024)		0.024							LEGOBench	MS-PR[‡]	Llama 2 7B
Singh et al. (2024)		0.077							LEGOBench	MS-PR[‡]	Llama 2 Chat 7B
Singh et al. (2024)		0.351							LEGOBench	MS-PR[‡]	Mistral 7B
Singh et al. (2024)	5.75	20.42							LEGOBench	MS-PR[‡]	Mistral Instr. 7B
Singh et al. (2024)		0.023							LEGOBench	MS-PR[‡]	Vicuna 7B
Singh et al. (2024)	1.49	10.87							LEGOBench	MS-PR[‡]	Zephyr Beta 7B
Singh et al. (2024)		0.014							LEGOBench	MS-PR[‡]	Llama 2 13B
Singh et al. (2024)		0.02							LEGOBench	MS-PR[‡]	Llama 2 Chat 13B
Singh et al. (2024)		0.06							LEGOBench	MS-PR[‡]	Vicuna 13B

Continued on next page.

	Micro			Macro			Part. Micro				
Reported In	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**F1**	**Dataset**	**Method**	**Experimental Setup**
Singh et al. (2024)	2.73	3.38							LEGOBench	MS-PR‡	Gemini Pro
Singh et al. (2024)	17.14	25.24							LEGOBench	MS-PR‡	GPT-4

* trained on ORKG-PwC-v6/v7 †SM, MLC, and EL are baseline methods, representing String Match, Multi-Label Classification, and Entity Linking, respectively. ‡Conditional on ⟨task, dataset, metric⟩. # REC, TAET, and Full refer to DocREC, DocTAET, and the Full Paper representations of the document, respectively. These are reported as part of an ablation study examining different document representations. For more details on these representations, see § 5.1.

Table 7: Summary of results for leaderboard tuple extraction, evaluated using variations of Micro and Macro Precision (**P**), Recall (**R**), and **F1** scores. Notations: **FS** = Few Shot, **ZS** = Zero Shot, **Instr.** = Instruction.