Zero-Shot Open-Schema Entity Structure Discovery

Anonymous ACL submission

Abstract

Entity structure extraction, which aims to extract entities and their associated attribute–value structures from text, is an essential task for text understanding and knowledge graph construction. Existing methods based on large language models (LLMs) typically rely heavily on predefined entity attribute schemas or annotated datasets, often leading to incomplete extraction results. To address these challenges, we introduce *Zero-Shot Open-schema Entity Structure Discovery* (ZOES), a novel approach to entity structure extraction that does not require any schema or annotated samples. ZOES operates via a principled mechanism of enrichment, refinement, and unification, based on the insight that an entity and its associated structure are mutually reinforcing. Experiments demonstrate that ZOES consistently enhances LLMs' ability to extract more complete entity structures across three different domains, showcasing both the effectiveness and generalizability of the method. These findings suggest that such an enrichment, refinement, and unification mechanism may serve as a principled approach to improving the quality of LLM-based entity structure discovery in various scenarios.

1 Introduction

Automatic mining of structured entity information is critical for knowledge discovery and management (Zhong et al., 2023a; Arsenyan et al., 2024). Prior works on entity information extraction—including relation extraction (Ding et al., 2024; Zhou et al., 2024; Zhang et al., 2025), entity typing (Onoe and Durrett, 2020; Tong et al., 2025), and named entity recognition (Li et al., 2020; Keraghel et al., 2024)—have primarily focused on extracting isolated aspects of entity knowledge. However, modeling only a single aspect of entity information may be insufficient for real-world applications (Jiao et al., 2023; Dagdelen et al., 2024).

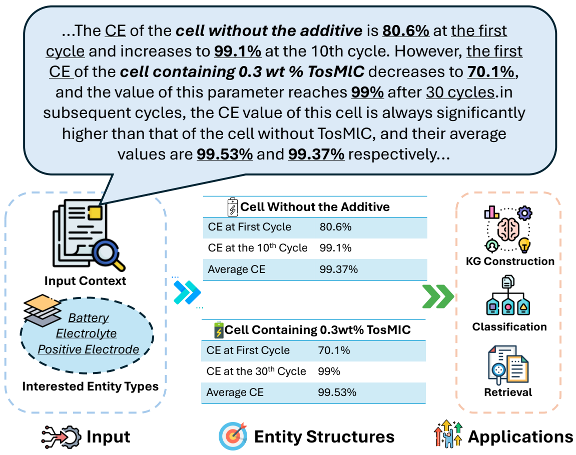

Figure 1: An example of the entity structure discovery task with applications. The figure depicts CEs of two discovered cells with their attributes and values organized as in the source passage from (Zhu et al., 2023).

For example, in the battery science domain, a battery's performance is determined by complex conditions (Zhou et al., 2023). As shown in Figure 1, even for the same battery, its "Coulombic Efficiency" (CE) value varies across different cycles. A single triplet (e.g., ⟨Cell Without the Additive, CE at First Cycle, 80.6% ⟩) conveys limited information about the battery's performance. In contrast, unifying performance across different conditions into a structured representation provides a clearer and more comprehensive view. Therefore, there is a need for a unified representation of entity information—one that integrates multiple aspects rather than focusing on a single one (Lu et al., 2023).

Recently, closed-schema entity structure extraction has been proposed to unify various aspects of entity information under predefined type schemas, where each entity type is associated with a fixed set of attributes (Zhong et al., 2023b; Wu et al., 2024). The goal is to extract structured entities represented as the entity along with a set of ⟨attribute, value⟩ pairs. By combining with the entity name to form

⟨entity, attribute, value⟩ triplet, it can capture a specific property of the entity, as illustrated in Figure 1. However, like other closed-schema information extraction tasks (Li et al., 2021; Zhou et al., 2023), entity type schemas confine the extraction on a limited set of attributes, which fail to capture diverse and unseen attributes in fast-evolving real-world scenarios (Pai et al., 2024).

To enable entity structure extraction to capture more diverse and dynamic information, we extend traditional closed-schema entity structure extraction to an open information extraction setting (Mausam, 2016), which we term **Open-Schema Entity Structure Discovery** (OpenESD). In OpenESD, we want to identify entities within user interests *and* their ⟨attribute, value⟩ structures *without* any predefined attribute sets as a schema. OpenESD can benefit many downstream tasks such as information retrieval (Kang et al., 2024) and question answering (Edge et al., 2025; Gutiérrez et al., 2025; Jiang et al., 2025).

With an open-schema setting, OpenESD goes beyond straightforward extraction: it demands discovering (Jiao et al., 2023), organizing (Wu et al., 2024), and inferring (Ding et al., 2024) the most appropriate attributes and values for each entity.

Large language models (LLMs) with extensive parametric knowledge have demonstrated promising performance in open information extraction (Jiao et al., 2022; Lu et al., 2023), offering a promising solution for OpenESD. However, fully harnessing this capability remains challenging. (i) **Extraction Coverage**: An LLM tends to capture coarse-grained facts that are more frequent in its parametric knowledge while missing rare, fine-grained information from the context. (ii) **Extraction Granularity**: When the context contains rich details, LLMs may fail to identify the appropriate level of granularity for representing the extracted information, resulting in incomplete or ambiguous structures. For example, as illustrated in Figure 1, if the extracted "CE" attributes fail to capture contextual conditions, multiple "CE" values may be incorrectly mapped to the same attribute, leading to inaccurate results.

To enhance LLMs' capability on OpenESD, we introducing ZOES, a **zero-shot open-schema entity structure discovery** framework. By employing a principled mechanism of enrichment, refinement, and unification, ZOES effectively extracts structured entity information without supervision. Specifically, ZOES starts with the LLM's zero-shot ⟨attribute, value⟩ triplets results, then gradually discovers new triplets to enrich it. Next, ZOES leverages mutual dependencies among triplet elements to identify and refine inferior triplets. Finally, the refined triplets are aggregated into entity structures as coherent representations of the entities based on user interest.

We evaluate ZOES using different backbone models on one long-tail domain: `Battery Science` and two general domains: `Economics` and `Politics`. The results demonstrate that ZOES can consistently outperform baselines with different backbone models in all domains. ZOES achieves an absolute improvement of $+10.64\%$ in the F1 score. These results demonstrate the effectiveness and generalizability of our method for OpenESD.

Our contributions are summarized as follows.

1. We introduce open-schema entity structure discovery, a task to automatically identify entities within user interest along with their contextual ⟨attribute, value⟩ structures without any predefined schema, which can benefit several knowledge intensive tasks.
2. We propose ZOES, a zero-shot open-schema entity structure discovery method. By enrich-refine-unify strategy, ZOES substantially improves LLMs' performance on OpenESD.
3. We construct labeled datasets on three very different domains and comprehensively evaluate ZOES and baselines to further studies LLMs' capabilities on OpenESD.

2 Related Work

2.1 Open Information Extraction

Open Information Extraction (OpenIE) aims to extract structured information from unstructured text without relying on predefined schemas (Zhou et al., 2022; Pai et al., 2024). Early OpenIE relied on rule-based methods (Del Corro and Gemulla, 2013; Mausam, 2016), sequence labeling (Ro et al., 2020; Vasilkovsky et al., 2022; Yu et al., 2021), or sequence-to-sequence models (Kolluru et al., 2022) to extract relational triplets from individual sentences. However, sentence-level relation extractions cannot capture cross-sentence relational information (Dunn et al., 2022; Wu et al., 2024), which leads to low information extraction coverage and limited representation quality (Li et al., 2021; Dagdelen et al., 2024).

Recent advances in OpenIE focus on leveraging LLMs to perform more expressive and instruction-following extractions (Jiao et al., 2023; Qi et al., 2024). These models support more flexible and user-guided information extraction, moving beyond fixed triplet formats toward on-demand schemas (Qi et al., 2024). While these approaches significantly improve the coverage and adaptability of OpenIE, they typically require substantial annotated training data or task-specific instruction tuning (Lu et al., 2023), which constrains their applicability in low-resource or specialized domains (Wei et al., 2023). Compared with previous works, ZOES focuses on a generalizable approach to guide LLMs to unify document-level entity information into structured representations by leveraging internal structural consistency, rather than relying on extensive training or annotations.

2.2 Zero-shot Relation Extraction

Zero-shot relation extraction (ZSRE) aims to identify semantic relations between entities without relying on labeled training instances (Levy et al., 2017). Prior work has predominantly approached this task by leveraging semantic representations to generalize to unseen relations (Chen and Li, 2021; Tran et al., 2022; Zhao et al., 2023). For example, Chen and Li (2021) proposed ZS-BERT, a supervised model that learns relation embeddings from attribute descriptions. Similarly, Zhao et al. (2023) introduced a fine-grained matching framework that integrates both entity and context embeddings to enhance zero-shot prediction. However, such embedding-based methods are sensitive to the exact wording of relation labels, limiting their robustness and generalizability in real-world settings.

More recently, LLMs have enabled a new paradigm in zero-shot relation extraction (Li et al., 2023; Xue et al., 2024; Zhou et al., 2024). One line of work explores using LLMs to generate relational statements directly from entity mentions, rather than extracting from predefined relation schemas or sentence-level contexts (Jiang et al., 2024; Ding et al., 2024). For instance, Ding et al. (2024) leverage LLMs' implicit understanding of entity types to generate topic-specific relations by aggregating corpus-level evidence. While these methods demonstrate strong generalization capabilities, they often produce high-level or generic relations. Our work explores utilizing LLMs to extract highly contextualized entity structures directly from input context without external knowledge.

3 Method

In this section, we start with the task formulation of open-schema entity structure discovery, and then delve into ZOES, a three stages approach for performing the task of OpenESD in detail. An illustrated overview of ZOES is in Figure 2.

3.1 Task Formulation

Open-schema entity structure discovery aims to automatically identify entities and their corresponding structures, from an input document and a given set of entity types of interest, without relying on any pre-defined schemas (e.g., pre-defined attribute names). The structure of each entity is represented as a set of ⟨attribute, value⟩ pairs, where entities and their associated structures are derived from the document. As an example, Figure 1 contains a battery science domain document discussing multiple properties regarding the entities "Cell Without the Additive" and "Cell Containing 0.3wt% TosMIC". The discovered entity structures should organize those properties as a set of attribute-value pairs, like attribute: "CE at First Cycle" with value: "80.6%" for "Cell Without the Additive".

Formally, given a document d and a set of entity types of interest $\mathcal{T}$, the goal is to identify a set of entities $\mathcal{E}$ within $\mathcal{T}$ such that $\mathcal{E} = \{e_1, \ldots, e_m\}$ and extract the structure of each entity. For an entity $e_i \in \mathcal{E}$, let $A_i = \{a_{i,1}, \ldots, a_{i,m}\}$ be the set of attributes and $V_i = \{v_{i,1}, \ldots, v_{i,m}\}$ be the corresponding set of values. We then define the structure S_i as the mapping $S_i : A_i \to V_i$, which can be represented alternatively as

$$\mathcal{S}_i = \big\{ (a, v) \mid a \in A_i,\, v \in V_i \big\}.$$

Each pair $\big(a, v\big)$ in $\mathcal{S}_i$ captures a property of e_i that can be inferred from the document d.

3.2 Triplet Candidates Extraction

Zero-shot triplet extraction using LLMs often suffers from limited knowledge coverage, as LLMs tend to prioritize extracting explicitly mentioned and high-frequency attribute-value pairs. Edge et al. (2025) attempt to improve coverage by prompting LLMs for multiple extraction rounds. However, without targeted guidance, such multi-round generation frequently yields redundant or noisy triplets, while still failing to recover low-salience but semantically meaningful triplets.

To address this challenge, ZOES first induces root attributes from an LLM's initial extracted triplets

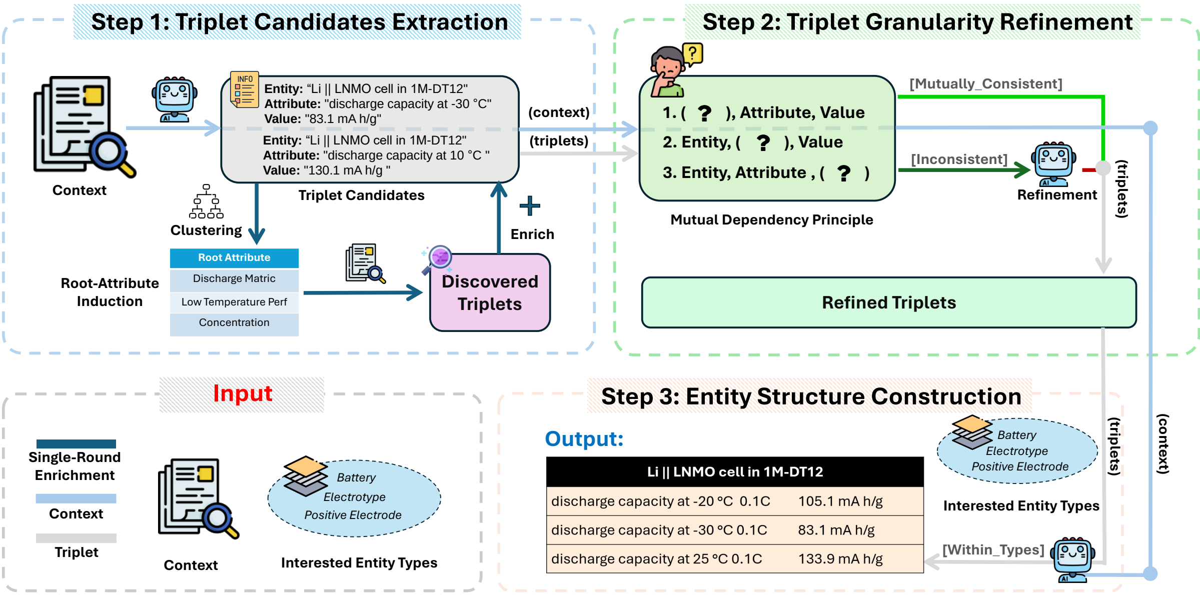

Figure 2: **Methodology Overview of ZOES.** ZOES operates in three stages: (1) **Triplet Candidates Extraction** expands the initial zero-shot EAV triplet set by leveraging generalized root attributes induced from initial extractions as guidance to uncover additional triplets; (2) **Triplet Granularity Refinement** applies the triplet mutual dependency principle to detect and revise under-specified or inconsistent triplets; and (3) **Entity Structure Construction** assembles refined triplets into entity structures, which are filtered based on user-specified target entity types.

$T_{initial}$. These root attributes serve as semantic guidance that clarify what kinds of values are valid or expected from the context, which assists the LLM to revisit the document context to discover missing triplets.

Root Attribute Induction. The initial zero-shot extraction yields a set of $\langle$entity, attribute, value$\rangle$ triplets $T_{initial}$, where some attributes are specific (e.g., *"CE at first cycle"*, *"initial CE"*). Such fine-grained attributes often correspond to only one triplet. In contrast, a general attribute such as *"Coulombic Efficiency"* can mapping to a set of potential values. We can utilize more general attributes to identify those previously missing values, thus identifying missing triplets.

Motivated by this observation, we induce *root attributes* that abstract over semantically similar attributes to further guide the triplet enrichment in the following stage. We first embed all extracted attributes using a dense encoder (Wang et al., 2022), and then we cluster them based on semantic similarity by agglomerative clustering (Ward Jr, 1963). This clustering step can groups attributes that express the same underlying general attribute. For each cluster, we prompt the LLM to summarize its members into a coarse-grained root attribute (e.g., *"Coulombic Efficiency"* from *"CE at first cycle"*, *"CE at the 10th cycle"*).

Value-Anchored Enrichment. Once root attributes are identified, we use them to guide the discovery of additional value mentions. For each root attribute, we prompt the LLM to revisit the document and list all corresponding values. This step often recovers contextually grounded values (e.g., *"higher"* a value comparing the CE among two cells) that align with the root attribute but were missed initially.

Although some entities may lack explicitly stated attribute–value structures in the context, each semantically meaningful value (e.g., "80.6%") should correspond to at least one valid triplet. Based on this intuition, each newly discovered value is treated as an anchor to elicit a missing triplet. We then prompt the LLM to infer the corresponding entity and attribute, constrained by the associated root attribute. This targeted prompting enables the recovery of under-expressed or indirectly stated facts, significantly improving extraction coverage.

By using root attributes as interpretable guides and values as anchors, this enrichment process helps the LLM uncover a more complete and semantically coherent $\langle$entity, attribute, value$\rangle$ triplet set T_{enrich}.

3.3 Triplets Granularity Refinement

Directly prompting LLMs to produce triplets in a zero-shot setting often yields suboptimal results to capture complex conditions, since LLMs lack an explicit understanding of the granularity required to represent entity structures unambiguously. To address this, we propose a refinement mechanism grounded in the **Mutual Dependency Principle**:

> For a triplet $t = \langle e, a, v \rangle$, we assume that appropriate granularity is achieved when any one component can be reliably inferred from the other two within the context d.

Based on this principle, given a triplet $t = \langle e, a, v \rangle$ from context d, we generate three questions, each aims to recover one component based on the other two and the context. Specifically, for each triplet $t_i = \langle e_i, a_i, v_i \rangle \in T_{\text{enrich}}$, we construct:

$$QA(t_i, d) = \Big\{ \langle q_c, \text{ans}_c \rangle \;\Big|\; \text{ans}_c \in \{e_i, a_i, v_i\}, \\ q_c \in \text{LLM}(t_i, d) \Big\}$$

For example, regarding a triplet $\langle$Cell without the Additive, CE, higher $\rangle$, can construct questions:

- *Which cell shows a higher CE?*
- *What is higher for the cell without the additive?*
- *What is the CE of the cell without the additive?*

The LLM is then prompted to answer these questions based on context d. We compare the predicted answer ans_p with the masked ground-truth component ans_c. A triplet is considered **mutually consistent** if all three components can be accurately recovered. Otherwise, it is flagged for refinement. For instance, if the original triplet is $\langle$Cell without the Additive, CE, higher $\rangle$, by giving only the entity and attribute, multiple values can be inferred from the context, which are not necessarily "higher". This indicates that the attribute lacks specificity and needs refinement. To perform refinement, we treat the value v_i as an anchor and prompt the LLM to revise the entity and attribute conditioned on v_i and context d.

This dependency-driven refinement helps identify and correct coarse or under-specified triplets, ensuring that only mutually-consistent triplets are retained. We denote the final set of refined triplets as T_{refine}, which serves as the input to the subsequent structure construction phase.

3.4 Entity Structure Construction

The final step of ZOES is to merge refined triplets into coherent entity structures, as illustrated in Figure 1. Since the refinement step (Section 3.3) utilizes the mutual dependency principle, the resulting triplets possess better granularity to accurately convey meaningful information unambiguously. To construct entity structures, we directly prompt the LLM with both the document context d and the refined triplet set T_{refine} to merge triplets discussing the same entity to form entity structures $\mathcal{E}_{\text{initial}}$.

Structure-Aware Filtering In real-world applications, users often have specific types of entities of interest, denoted as a target type set $\mathcal{T}$. For each structured entity $e_i \in \mathcal{E}_{\text{initial}}$, we use the LLM to determine whether it belongs to the desired types, based on its attributes, values, and the document context:

$$\text{LLM}(e_i, \mathcal{T} \mid d) \rightarrow \{\text{True}, \text{False}\}$$

This structure-aware filtering enables ZOES to utilize entity structures to augment entity names' semantics. In many domains, entity names alone are insufficient to determine their relevance or type. For instance, in battery science, entities such as "fluoroethylene carbonate" may not clearly indicate its entity types even with context. However, if we know it has an attribute as a function in battery electrolyte, the LLM can directly know its type is "electrolyte addictive". Finally, by construction and filtration, ZOES can produce contextually grounded entity structures $\mathcal{E}$ in a zero-shot setting.

4 Experiments

We begin with the experimental setup, including dataset construction, evaluation metrics, and implementation details. We then present our main results, followed by ablation studies evaluating the effectiveness of each component in ZOES.

4.1 Dataset Construction

We construct an entity structure extraction dataset spanning one long-tail domain, `Battery Science`, and two general domains, `Economics` and `Politics`. The dataset specifically focuses on evaluating two challenges of OpenESD: *extraction coverage* and *extraction granularity* introduced in Section 1. For each domain, the dataset contains a set of documents and a set of interested entity types. We will later release the dataset publicly. The statistics of the dataset can be found in Table 2.

Long-Tail Domain. For the `Battery Science` domain, we curate paragraphs from top-tier peer-

Model	Method	Battery Science			Economics			Politics		
		Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Llama-3.2-3B	Text2Triple (SFT)	0.2634	0.1718	0.2083	0.7312	0.6538	0.6903	0.8416	0.7852	0.8124
GPT-4o	CoT	0.6087	0.4275	0.5022	0.8880	0.6619	0.7585	0.8214	0.1593	0.2669
	Few-Shot	**0.7911**	0.4771	0.5952	**0.9046**	0.7149	0.7986	**0.9295**	0.6397	0.7579
	Zoes (Ours)	0.7758	**0.6844**	**0.7287**	0.8994	**0.9104**	**0.9049**	0.8534	**0.9007**	**0.8764**
GPT-4o-mini	CoT	0.5562	0.3779	0.4500	0.8493	0.5967	0.7010	0.5952	0.1155	0.1934
	Few-Shot	0.5102	0.3816	0.4367	**0.8657**	0.7352	0.7952	**0.8933**	0.6767	0.7700
	Zoes (Ours)	**0.5708**	**0.6441**	**0.6104**	0.8532	**0.8289**	**0.8409**	0.8374	**0.7852**	**0.8105**
Granite-8B	CoT	0.6149	0.3473	0.4439	0.7051	0.4236	0.5293	0.7241	0.0970	0.1711
	Few-Shot	**0.6579**	0.3817	0.4831	0.7398	0.5153	0.6074	0.7431	0.4341	0.5481
	Zoes (Ours)	0.5708	**0.5229**	**0.5458**	**0.8017**	**0.7821**	**0.7918**	**0.7790**	**0.8383**	**0.8076**

Table 1: Evaluation with user interested entity types across different backbone models and methods on Battery Science, Economics, and Politics. Bold numbers highlight the best results per backbone model in Battery Science.

Domain	#Documents	#Sentences	#(E, A, V)s
BatSci	20	197	428
Finance	50	195	491
Politics	50	208	433
Overall	120	675	1,289

Table 2: Dataset statistics across "Battery Science", "Finance", and "Politics" domains. "BatSci" stands for "Battery Science," and "(E, A, V)s" denotes $\langle$entity, attribute, value$\rangle$ triplets.

reviewed research articles that discuss the performance and applications of battery components. These paragraphs are characterized by diverse experimental conditions and frequent comparisons across similar components. Missing contextual conditions in such cases can result in misleading or contradictory information. Furthermore, the text contains domain-specific terminology and fine-grained technical descriptions, posing significant challenges for LLMs to accurately understand and extract entity structures. This domain exemplifies the long-tail scenario: high knowledge granularity, low representation in pretraining corpora, and substantial variance in how attributes are expressed.

General Domain. We collect paragraphs from mainstream news agencies, including The Economist, Fox News, CNN, and BBC, in the `Economics` and `Politics` domains to evaluate the methods' performance in general-purpose scenarios. In the Economics domain, the selected texts contain analyses with rich numerical data and fine-grained economic indicators, making it challenging for LLMs to identify and associate context-specific attribute–value pairs with the correct entities. For the Politics domain, all documents contain diverse entities whose attributes are scattered across sentences, posing challenges for extraction completeness. Successful extraction in this setting requires models to rely solely on contextual understanding to recognize entities and infer their corresponding attributes and values.

4.2 Evaluation

To comprehensively evaluate each method's ability to extract fine-grained information, we follow prior structured entity extraction work (Wu et al., 2024), reporting `Precision`, `Recall`, and `F1` scores at the $\langle$entity, attribute, value$\rangle$ triplet level. To ensure high-quality ground truth annotations, we adopt a pooling-based evaluation strategy: *aggregate all extracted triplets across methods and have experienced annotators from each domain validate them to construct the reference set.* Full details on the evaluation criteria and annotation process are provided in Appendix A.

Baselines. Since OpenESD requires contextual understanding to induce attributes from text—unlike traditional extractive information extraction tasks (Nasar et al., 2021; Zhou et al., 2024)—we evaluate LLM-based approaches under both training-based and training-free settings.

For the training-based setting, we report results from **Text2Triple** (Jiang et al., 2025), a 3B language model fine-tuned on a general-domain open triplet extraction dataset comprising 2 million instances curated using `Claude-Sonnet-3.5`.

For training-free methods, we consider three prompting strategies: **Chain-of-Thought (CoT)** prompting (Wei et al., 2022), **Few-Shot** prompting (Brown et al., 2020), and our proposed method ZOES. All three are evaluated using the following backbone models: GPT-4o (OpenAI et al., 2024), GPT-4o-mini, and Granite-8B (Granite Team, 2024). Prompting templates are provided in Table 5.

4.3 Main Results.

Table 1 summarizes the performance of all evaluated methods across three domains: Battery Science, Economics, and Politics with three backbone models. We have the following observations: ZOES consistently achieves the highest F1 scores across all domains and backbone models, outperforming both **CoT** and **Few-Shot** prompting. This highlights the effectiveness and generalizability of ZOES in extracting accurate and comprehensive entity structures without relying on annotated data. However, we also observe that ZOES sometimes exhibits lower precision compared to other baselines. This may be because ZOES's enrichment module 3.2 not only recovers potentially missed extractions but also introduces noise into the results. We further analyze the contribution of each module of ZOES in ablation studies 4.4.

Few-shot prompting generally improves performance, surpassing CoT in most cases in terms of precision, recall, and F1 score. This confirms the importance of in-context demonstrations in helping LLMs identify relevant attributes and values in open-schema settings. However, in the Battery Science domain, the improvement of few-shot prompting on recall is less pronounced, suggesting that in long-tail or highly specialized domains, few-shot examples may be insufficient for uncovering latent, context-dependent attributes—particularly when those attributes are nested within complex experimental conditions. These results highlight the benefit of ZOES 's approach: abstracting attributes into coarse-grained representations to help LLMs uncover missing extractions, followed by a granularity refinement step to recover fine-grained contextual conditions.

While supervised fine-tuning can significantly enhance model performance on in-distribution data, such improvements often fail to generalize to unseen domains. In our experiments, Text2Triple (Jiang et al., 2025), a model fine-tuned on general domain, achieves strong performance in the Politics domain, with competitive scores in Precision, Recall, and F1. However, its effectiveness becomes less prominent in the Economics domain and drops substantially in the Battery Science domain. This degradation highlights the limited transferability of supervised approaches when faced with domain-specific or out-of-distribution contexts. In contrast, training-free methods, especially ZOES, demonstrate consistently robust performance across all domains, underscoring their adaptability and reliability in zero-shot settings.

4.4 Ablation Analysis

To evaluate the contributions of ZOES's core components, we conduct ablation studies by removing two key modules: (1) Value-Anchored Enrichment (cf. Section 3.2) and (2) Mutual Dependency-Based Triplet Refinement (cf. Section 3.3). We evaluate each variant using GPT-4o as the backbone model and report the results in Table 3.

Method	Precision	Recall	F1
ZOES	**0.8994**	**0.9104**	**0.9049**
w/o Enrich	0.8465	0.8758	0.8609
w/o Refine	0.8143	0.8839	0.8477

Table 3: Ablation results evaluated by Precision, Recall, and F1 on the Finance domain using GPT-4o as the backbone.

As shown in Table 3, removing either component consistently degrades ZOES's performance, demonstrating the effectiveness of each module's design. Specifically, the Mutual Dependency-Based Triplet Refinement module is responsible for correcting potentially incorrect or incomplete extraction results. Removing this module noticeably reduces precision, as the model tends to include overgeneralized or ambiguous triplets that may have been introduced by the enrichment module.

These results also show that enrichment and refinement collaboratively enhance ZOES's performance: the enrichment module increases extraction coverage by discovering previously missed information, though it may also yield incomplete results due to the subtlety of certain implicitly mentioned attributes. Meanwhile, the refinement module helps detect and revise ambiguous or partial extractions, thereby improving the quality of enrichment.

4.5 Coverage Win Rate

To assess extraction coverage across methods, we compute a coverage win rate for each backbone model (GPT-4o, GPT-4o Mini, Granite-8B) under three prompting strategies (CoT, Few-Shot, ZOES) on a per-document basis in the Economics domain.

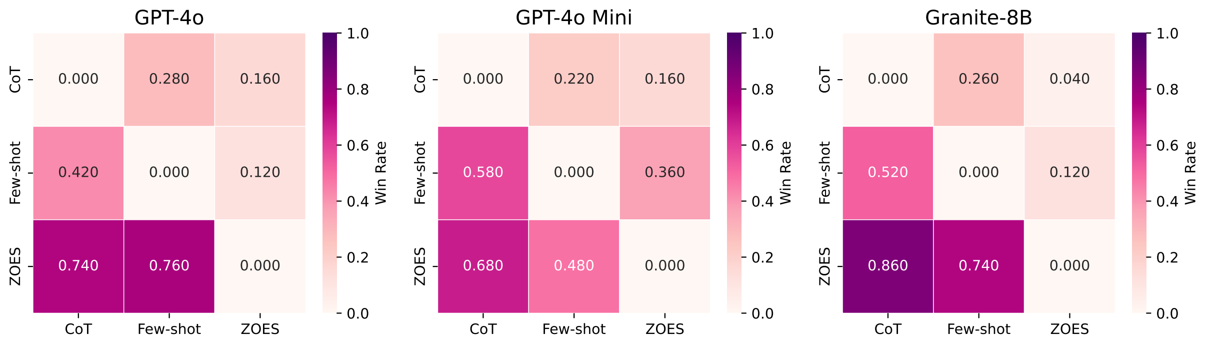

Figure 3: **Prompting-Based Extraction Coverage Win Rate** of different backbone models (GPT-4o, GPT-4o Mini, Granite-8B) using various prompting methods (CoT, Few-Shot, ZOES) in the Economics domain. Each heat map shows the pairwise win rate between methods, where the value in row i, column j represents the proportion of test instances for which method i extracts more correct triplets than method j. For example, with GPT-4o, ZOES outperforms Chain-of-Thought prompting in 74% of instances (win rate = 0.740).

For each document, two annotators independently compare the extraction results of every method pair. If both annotators agree that one method extracts more complete and informative triplets than the other, it is counted as a win; otherwise, the comparison is marked as a tie. As shown in Figure 3, ZOES consistently achieves higher win rates compared to both CoT and Few-Shot prompting across all models. This indicates that even without training data, ZOES is capable of capturing more comprehensive information from diverse contexts, reinforcing its effectiveness in zero-shot open-schema entity structure discovery.

Method	**Extracted Entity Structures (Triplets)**
Few-shot	(Toyota, largest automaker, world), (Toyota, annual car sales in US, 2.3 million), (Toyota, profitability status, profitable), (Toyota, reputation among analysts, one of the best-run companies in global auto industry)
ZOES	(Toyota, Cars Sold in the US Last Year, 2.3M), (Toyota, Close Collaborators, Subaru & Mazda), (Toyota, Cars manufactured Outside US, 1M), (Toyota, Position Among Automakers, World's Largest), (Toyota, Profitability, Profitable)

Table 4: Example extracted results on a sample document from the Economics domain for "Toyota," by ZOES and Few-Shot methods using Granite-8B.

4.6 Case Studies

As shown in Table 4, ZOES produces more complete and contextually faithful extractions than Few-Shot prompting. First, ZOES captures more fine-grained and semantically rich attributes (e.g., "Cars Sold in the US Last Year", "Close Collaborators") compared to the relatively generic expressions extracted by Few-Shot (e.g., "annual car sales in US"). This improvement stems from ZOES's mutual-dependency-based triplet refinement, which detects and refines ambiguous triplets. Second, ZOES demonstrates better coverage by identifying additional informative triplets that are absent in Few-Shot results, (e.g., "(Toyota, Cars manufactured Outside US, 1M)") This is enabled by the value-anchored enrichment mechanism, which revisits the document to recover missing triplets under guided root attributes.

5 Conclusions

We introduce ZOES, a zero-shot, training-free framework for *open-schema entity structure discovery* without relying on predefined schemas or annotated data. ZOES achieves high-quality entity structure extraction across both long-tail and general domains. Extensive experiments demonstrate that ZOES not only substantially improves the performance of smaller language models in a zero-shot setting, but also outperforms baselines across three diverse domains. Our findings suggest that explicitly structuring the entity discovery process rather than relying on static prompting alone offers a robust and principled approach to information extraction in long-tail, open-world scenarios. We believe ZOES is good experimental evidence for schema-free knowledge extraction with LLMs and provides a foundation for future research in context-grounded entity understanding.

Limitations

This work introduces ZOES, a training-free zero-shot entity structure discovery method, and develops a dataset on three distinct domains to evaluate its performance against zero-shot and supervised baselines. We discuss the following limitations:

Computational Efficiency. Although ZOES substantially improves LLM performance on open-schema entity structure extraction, it involves multiple rounds of generation, enrichment, and refinement. This pipeline process increases computational cost and inference time, which may hinder scalability. One potential research direction is to utilize ZOES extraction results as demonstrations for LLMs' few-shot learning on open-schema entity structure discovery.

Evaluation Metrics. Our evaluation relies on human-annotated reference triplets and a weighted scoring function to assess the correctness and completeness of extracted structures. While this ensures high-quality assessment, the reliance on manual annotation can introduce subjectivity and may not scale efficiently to broader domains. Future work could explore more automated and domain-agnostic evaluation strategies to improve scalability and reproducibility.

Ethical Statement

We uphold ethical principles throughout the design, development, and evaluation of ZOES. The dataset used in this work was curated with careful attention to exclude any personally identifiable or sensitive information. All documents included were collected in accordance with their respective licensing agreements and terms of use.

Human-annotated test data were collected with informed consent, following ethical research guidelines. To promote fairness and reduce potential bias, we curated a diverse dataset across three domains and verified that entity types and contextual structures were broadly representative.

References

Vahan Arsenyan, Spartak Bughdaryan, Fadi Shaya, Kent Wilson Small, and Davit Shahnazaryan. 2024. Large language models for biomedical knowledge graph construction: Information extraction from EMR notes. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pages 295–317, Bangkok, Thailand. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chih-Yao Chen and Cheng-Te Li. 2021. ZS-BERT: Towards zero-shot relation extraction with attribute representation learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3470–3479, Online. Association for Computational Linguistics.

John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S. Rosen, Gerbrand Ceder, Kristin A. Persson, and Anubhav Jain. 2024. Structured information extraction from scientific text with large language models. *Nature Communications*, 15(1):1418.

Luciano Del Corro and Rainer Gemulla. 2013. Clausie: clause-based open information extraction. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, page 355–366, New York, NY, USA. Association for Computing Machinery.

Linyi Ding, Jinfeng Xiao, Sizhe Zhou, Chaoqi Yang, and Jiawei Han. 2024. Topic-oriented open relation extraction with a priori seed generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13834–13845, Miami, Florida, USA. Association for Computational Linguistics.

Alexander Dunn, John Dagdelen, Nicholas Walker, Sanghoon Lee, Andrew S. Rosen, Gerbrand Ceder, Kristin Persson, and Anubhav Jain. 2022. Structured information extraction from complex scientific text with fine-tuned large language models. *Preprint*, arXiv:2212.05238.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2025. From local to global: A graph rag approach to query-focused summarization. *Preprint*, arXiv:2404.16130.

IBM Granite Team. 2024. Granite 3.0 language models.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. 2025. From rag to memory: Non-parametric continual learning for large language models. *Preprint*, arXiv:2502.14802.

Pengcheng Jiang, Lang Cao, Ruike Zhu, Minhao Jiang, Yunyi Zhang, Jimeng Sun, and Jiawei Han. 2025. Ras: Retrieval-and-structuring for knowledge-intensive llm generation. *arXiv preprint arXiv: 2502.10996*.

Pengcheng Jiang, Jiacheng Lin, Zifeng Wang, Jimeng Sun, and Jiawei Han. 2024. GenRES: Rethinking evaluation for generative relation extraction in the era of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2820–2837, Mexico City, Mexico. Association for Computational Linguistics.

Yizhu Jiao, Sha Li, Yiqing Xie, Ming Zhong, Heng Ji, and Jiawei Han. 2022. Open-vocabulary argument role prediction for event extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5404–5418, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yizhu Jiao, Ming Zhong, Sha Li, Ruining Zhao, Siru Ouyang, Heng Ji, and Jiawei Han. 2023. Instruct and extract: Instruction tuning for on-demand information extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10030–10051, Singapore. Association for Computational Linguistics.

SeongKu Kang, Yunyi Zhang, Pengcheng Jiang, Dongha Lee, Jiawei Han, and Hwanjo Yu. 2024. Taxonomy-guided semantic indexing for academic paper search. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7169–7184, Miami, Florida, USA. Association for Computational Linguistics.

Imed Keraghel, Stanislas Morbieu, and Mohamed Nadif. 2024. A survey on recent advances in named entity recognition. *arXiv preprint arXiv:2401.10825*.

Keshav Kolluru, Muqeeth Mohammed, Shubham Mittal, Soumen Chakrabarti, and Mausam. 2022. Alignment-augmented consistent translation for multilingual open information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2502–2517, Dublin, Ireland. Association for Computational Linguistics.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, Vancouver, Canada. Association for Computational Linguistics.

Guozheng Li, Peng Wang, and Wenjun Ke. 2023. Revisiting large language models as zero-shot relation extractors. *arXiv preprint arXiv:2310.05028*.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2020. A survey on deep learning for named entity recognition. *IEEE transactions on knowledge and data engineering*, 34(1):50–70.

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. *arXiv preprint arXiv:2104.05919*.

Keming Lu, Xiaoman Pan, Kaiqiang Song, Hongming Zhang, Dong Yu, and Jianshu Chen. 2023. PIVOINE: Instruction tuning for open-world entity profiling. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15108–15127, Singapore. Association for Computational Linguistics.

Mausam Mausam. 2016. Open information extraction systems and downstream applications. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 4074–4077. AAAI Press.

Zara Nasar, Syed Waqar Jaffry, and Muhammad Kamran Malik. 2021. Named entity recognition and relation extraction: State-of-the-art. *ACM Computing Surveys (CSUR)*, 54(1):1–39.

Yasumasa Onoe and Greg Durrett. 2020. Fine-grained entity typing for domain independent entity linking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8576–8583.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Liu Pai, Wenyang Gao, Wenjie Dong, Lin Ai, Ziwei Gong, Songfang Huang, Li Zongsheng, Ehsan Hoque, Julia Hirschberg, and Yue Zhang. 2024. A survey on open information extraction from rule-based model to large language model. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9586–9608, Miami, Florida, USA. Association for Computational Linguistics.

Yunjia Qi, Hao Peng, Xiaozhi Wang, Bin Xu, Lei Hou, and Juanzi Li. 2024. ADELIE: Aligning large language models on information extraction. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7371–7387, Miami, Florida, USA. Association for Computational Linguistics.

Youngbin Ro, Yukyung Lee, and Pilsung Kang. 2020. Multi^2OIE: Multilingual open information extraction based on multi-head attention with BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1107–1117, Online. Association for Computational Linguistics.

Zeliang Tong, Zhuojun Ding, and Wei Wei. 2025. EvoPrompt: Evolving prompts for enhanced zero-shot named entity recognition with large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5136–5153, Abu Dhabi, UAE. Association for Computational Linguistics.

Van-Hien Tran, Hiroki Ouchi, Taro Watanabe, and Yuji Matsumoto. 2022. Improving discriminative learning for zero-shot relation extraction. *Proceedings of the 1st Workshop on Semiparametric Methods in NLP: Decoupling Logic from Knowledge*.

Michael Vasilkovsky, Anton Alekseev, Valentin Malykh, Ilya Shenbin, Elena Tutubalina, Dmitriy Salikhov, Mikhail Stepnov, Andrei Chertok, and Sergey Nikolenko. 2022. DetIE: Multilingual Open Information Extraction Inspired by Object Detection. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301):236–244.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2023. Zero-shot information extraction via chatting with chatgpt. *ArXiv*, abs/2302.10205.

Haolun Wu, Ye Yuan, Liana Mikaelyan, Alexander Meulemans, Xue Liu, James Hensman, and Bhaskar Mitra. 2024. Learning to extract structured entities using language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6817–6834, Miami, Florida, USA. Association for Computational Linguistics.

Lilong Xue, Dan Zhang, Yuxiao Dong, and Jie Tang. 2024. Autore: Document-level relation extraction with large language models. *arXiv preprint arXiv:2403.14888*.

Bowen Yu, Yucheng Wang, Tingwen Liu, Hongsong Zhu, Limin Sun, and Bin Wang. 2021. Maximal clique based non-autoregressive open information extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9696–9706, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Fu Zhang, He Liu, Zehan Li, and Jingwei Cheng. 2025. CE-DA: Custom embedding and dynamic aggregation for zero-shot relation extraction. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 9814–9823, Abu Dhabi, UAE. Association for Computational Linguistics.

Jun Zhao, WenYu Zhan, Xin Zhao, Qi Zhang, Tao Gui, Zhongyu Wei, Junzhe Wang, Minlong Peng, and Mingming Sun. 2023. RE-matching: A fine-grained semantic matching method for zero-shot relation extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6680–6691, Toronto, Canada. Association for Computational Linguistics.

Lingfeng Zhong, Jia Wu, Qian Li, Hao Peng, and Xindong Wu. 2023a. A comprehensive survey on automatic knowledge graph construction. *ACM Comput. Surv.*, 56(4).

Ming Zhong, Siru Ouyang, Minhao Jiang, Vivian Hu, Yizhu Jiao, Xuan Wang, and Jiawei Han. 2023b. ReactIE: Enhancing chemical reaction extraction with weak supervision. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12120–12130, Toronto, Canada. Association for Computational Linguistics.

Shaowen Zhou, Bowen Yu, Aixin Sun, Cheng Long, Jingyang Li, and Jian Sun. 2022. A survey on neural open information extraction: Current status and future directions. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 5694–5701. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Sizhe Zhou, Suyu Ge, Jiaming Shen, and Jiawei Han. 2023. Corpus-based relation extraction by identifying and refining relation patterns. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 20–38. Springer.

Sizhe Zhou, Yu Meng, Bowen Jin, and Jiawei Han. 2024. Grasping the essentials: Tailoring large language models for zero-shot relation extraction. *Preprint*, arXiv:2402.11142.

Jin Zhu, Mingyu Zhang, Yunyan Gai, Ronghua Zeng, Yuepeng Cai, and Dongsheng Lu. 2023. Quickly form stable cathode/electrolyte interface of lini0.5mn1.5o4 (lnmo)/graphite high-voltage lithium ion cells by using tosylmethyl isocyanide (tosmic) as electrolyte additive. *Journal of Power Sources*, 576:233227.

A Evaluation

A.1 Evaluation Metrics

Let each domain's dataset be $\mathcal{D} = \{d_1, \ldots, d_{|\mathcal{D}|}\}$. For each document $d \in \mathcal{D}$, let P_d denote the set of predicted triples and G_d denote the set of ground-truth triples.

Each predicted triple $t \in P_d$ is scored by human annotators using the following scoring function $S(t)$, which measures the correctness and completeness of the extracted structure:

- $S(t) = 0$, if the triple is **incorrect**, or if the entity is not of an *interested type*.
- $S(t) = 0.5$, if the triple is **correct but incomplete**, e.g., the entity or value is only partially captured.
- $S(t) = 1$, if the triple is both **correct and complete**, with all components (entity, attribute, value) accurately captured.

To evaluate overall performance, we aggregate the scores across all documents. Define:

$$P = \bigcup_{d=1}^{|\mathcal{D}|} P_d \quad \text{and} \quad G = \bigcup_{d=1}^{|\mathcal{D}|} G_d.$$

We compute the evaluation metrics as:

$$\text{Precision} = \frac{\sum_{d=1}^{|\mathcal{D}|} \sum_{t \in P_d} S(t)}{\sum_{d=1}^{|\mathcal{D}|} |P_d|} \tag{1a}$$

$$\text{Recall} = \frac{\sum_{d=1}^{|\mathcal{D}|} \sum_{t \in P_d} S(t)}{\sum_{d=1}^{|\mathcal{D}|} |G_d|} \tag{1b}$$

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1c}$$

Precision measures the proportion of predicted triples that are judged as correct or partially correct. High precision indicates the model produces relevant and accurate triples, minimizing hallucinated or noisy outputs.

Recall quantifies how many ground-truth triples were successfully recovered. High recall implies strong extraction coverage over the true structured information.

F1 Score is the harmonic mean of precision and recall, balancing both correctness and coverage.

A.2 Human Annotation Protocol

To ensure rigorous evaluation, we divided the annotation task into two teams based on domain expertise:

Battery Science Domain. Two domain-expert researchers with Ph.D. degrees in science fields were recruited.:

- One annotator collected all baseline outputs and corrected extraction errors to construct the ground-truth triplets.
- The another annotator independently received anonymized extraction results from each method and judged them as *correct*, *partially correct*, or *incorrect* using the scoring rubric.

General Domain. Three annotators participated:

- A master's and an undergraduate student in computer science collaboratively constructed ground-truth triples from model outputs, following the same procedure.
- A third annotator (a senior undergraduate student) independently evaluated the model predictions in a blind review setting using the scoring function.

This process ensures that the evaluation is both context-sensitive and unbiased.

B Prompting Templates & Pseudocode of ZOES

Table 5 lists all prompting templates used in this study. For completeness, we also include the pseudocode of ZOES in Algorithm 1.

Prompt Name	**Prompt Template**
0-shot Triplet Extraction	You are an expert in information extraction. Extract all (entity, attribute, value) triplets from the document. Here is the Provided Document: [document]
0-shot Root Attribute Induction	You are a helpful information extraction assistant. Can you summarize a category name for the following values?
0-shot Value Extraction	You are a helpful information extraction assistant. Can you extract all values (exact text spans, with units) under [document] for each attribute in [root attribute]?
0-shot Value-Guided Triplet Extraction	You are an expert information extraction assistant. Given Document: [document] and value types, extract all values (exact text spans, with units) under each type.
Mutual Dependency QA (Question Generation)	You are a helpful question answering assistant. Given a <entity, attribute, value> triplet, generate three questions where each question asks for one component using only the other two as context. Do not infer or hallucinate new information.
Mutual Dependency QA (Question Answering)	You are a helpful question answering assistant. Please answer the following questions using answers extracted from the context. Context: [context] Question 1: Q_entity Question 2: Q_attribute Question 3: Q_value
Triplet Refinement	There is a <entity, attribute, value> triplet extracted from the context. The original triplet may cause ambiguity due to an incomplete entity or a non-informative attribute. Refine the given triplet by extracting exact information from the context, such that the attribute is a clear property of the entity. Context: [context] Triplet: <entity, attribute, value>
Entity Structure Construction	For a given list of (entity, attribute, value) triplets and a context, merge triplets referring to the same entity into structured objects. Follow this format: "entity name": "attribute": "value", ..., ... Context: [document] Triplets: [triplets]
Entity Type Filtration	You are a helpful assistant. For a given entity with its attribute and values, can you decide whether the entity belongs to any given entity types based on the context. The given context is: [Context]. The given triplets are [Triplets]. The given entity types are: [Entity Type]. Response “Yes” or “No”.
Chain-of-Thought Triplet Extraction	You are an expert in information extraction. Instructions: (1) Identify all precise entities of types in [T] that have associated characteristics. (2) For each entity, extract: - Entity: The name or title - Attribute: The key property - Value: The associated value (numerical, adjective, or noun phrase) Formatting: - Output only results - Format exactly: [entity, attribute, value] Document: [document]
Few-Shot Triplet Extraction	You are an expert in information extraction. Instructions: same as Chain-of-Thought Triplet Extraction. In addition, you are given: Demonstrations: [Demonstrations] Document: [document]

Table 5: Prompt templates used in this work. `[ ]` and `< >` denote placeholders.

Algorithm 1: ZOES: Zero-Shot Open-Schema Entity Structure Discovery

Input: Document d, Target entity types $\mathcal{T}$
Output: Structured entities $\mathcal{E}$

Step 1: Triplet Candidates Extraction

```
T_init ← LLM_ZeroShotExtract(d)
E_emb ← {f(t) | t ∈ T_init} // Embed triplets
C ← AgglomerativeClustering(E_emb, α)
R ← ∅
foreach C_i ∈ C do
    r_i ← LLM_SummarizeAttributes(C_i)
    R ← R ∪ {r_i}
T_enrich ← T_init
foreach r ∈ R do
    V_r ← LLM_ExtractValues(r, d)
    foreach v ∈ V_r do
        t_new ← LLM_InferTripletByValue(v, r, d)
        if t_new ≠ ∅ then
            T_enrich ← T_enrich ∪ {t_new}
```

Step 2: Triplet Granularity Refinement

```
T_refine ← ∅
foreach t = ⟨e, a, v⟩ ∈ T_enrich do
    is_consistent ← True
    foreach c ∈ {e, a, v} do
        q_c ← GenerateQuestion(t \ {c})
        a_c ← LLM_Answer(q_c, d)
        if a_c ≠ c then
            is_consistent ← False; break
    if is_consistent then
        T_refine ← T_refine ∪ {t}
    else
        t' ← LLM_RefineTriplet(v, d)
        if t' ≠ ∅ then
            T_refine ← T_refine ∪ {t'}
```

Step 3: Entity Structure Construction

```
E_init ← LLM_ConstructEntities(T_refine, d)
E ← ∅
foreach e ∈ E_init do
    if LLM_IsTypeMatch(e, T, d) then
        E ← E ∪ {e}
return E
```