LLMs4OL 2025: The 2nd Large Language Models for Ontology Learning Challenge at the 24th ISWC
LLMs4OL 2025 Task Participant Long Papers
https://doi.org/10.52825/ocp.v6i.2900

Published: 01 Oct. 2025

silp_nlp at LLMs4OL 2025 Tasks A, B, C, and D: Clustering-Based Ontology Learning Using LLMs

Pankaj Kumar Goyal[1], Sumit Singh[1,2,*], and Uma Shanker Tiwary[1]

[1]Indian Institute of Information Technology, Allahabad
[2]UPES University, Dehradun
*Correspondence: Sumit Singh, sumitrsch@gmail.com

Abstract. This paper presents the participation of the silp_nlp team in the LLMs4OL 2025 Challenge, where we addressed four core tasks in ontology learning: Text2Onto (Task A), Term Typing (Task B), Taxonomy Discovery (Task C), and Non-Taxonomic Relation Extraction (Task D). Building on our experience from the first edition, we proposed a clustering-enhanced methodology grounded in large language models (LLMs), integrating domain-adapted transformer models such as pranav-s/MaterialsBERT, dmis-lab/biobert-v1.1, and proprietary LLMs from Grok. Our framework combined lexical and semantic clustering with adaptive prompting to tackle entity and type extraction, semantic classification, hierarchical structure discovery, and complex relation modeling. Experimental results across 18 subtasks highlight the strength of our approach, particularly in blind and zero-shot scenarios. Notably, our model achieved multiple first-rank scores in taxonomy discovery and non-taxonomic relation extraction subtasks, validating the efficacy of clustering when coupled with semantically specialized LLMs. This work demonstrates that clustering-driven, LLM-based approaches can advance robust and scalable ontology learning across diverse domains.

Keywords: Ontology Learning, Large Language Models, Prompt Engineering, Clustering, Knowledge Representation

1. Introduction

The first iteration of the Large Language Models for Ontology Learning (LLMs4OL) Challenge marked a significant step toward leveraging large language models (LLMs) for automated ontology learning (OL). It demonstrated the potential of LLMs in extracting, classifying, and structuring domain-specific knowledge. The challenge included three core tasks: Term Typing, Taxonomy Discovery, and Non-Taxonomic Relation Extraction, and was evaluated in both few-shot and zero-shot settings. Participants explored a wide range of strategies, including prompt engineering, fine-tuning, and hybrid models that integrated LLMs with rule-based and retrieval-augmented techniques. Code of this work is available here[1] . Results highlighted that while LLMs perform well on hierarchical tasks like Term Typing and Taxonomy Discovery, they struggle with semantically

[1]Link of code

complex relation extraction, where hybrid or knowledge-enriched methods yield better performance.

Building upon our previous participation, we took part in the second iteration of the LLMs4OL Challenge [1], which introduced a more comprehensive benchmark composed of four tasks: (A) Text2Onto, (B) Term Typing, (C) Taxonomy Discovery, and (D) Non-Taxonomic Relation Extraction. These tasks together aimed to facilitate the transition from unstructured text to structured ontologies, encompassing entity and class extraction, semantic typing, hierarchical classification, and semantic relation modeling.

In this work, we propose a clustering-based methodology that leverages the representational strengths of state-of-the-art LLMs to address the complexity of these ontology learning tasks. Specifically, we employed domain-specialized transformer models such as `pranav-s/MaterialsBERT` [2], `dmis-lab/biobert-v1.1` [3], and LLMs from `grok.co` and `grok.com` [4]. These models were selected for their domain alignment with the sub-tasks of the challenge, allowing us to form semantically coherent clusters of terms and types. Our approach aimed to bridge lexical variation and domain-specific semantics by combining deep contextual embeddings with unsupervised clustering and adaptive prompting strategies.

The results across multiple subtasks confirm that clustering-driven representations, powered by specialized LLMs, can effectively enhance performance in both taxonomic and non-taxonomic relation inference. Furthermore, our comparative evaluation across biomedical, material science, and environmental datasets illustrates the adaptability of the proposed framework for diverse ontology learning domains.

Details of the primary tasks: Text2Onto, Term Typing, Taxonomy Discovery, and Non-Taxonomic Relation Extraction are described below.

1.1 Task A: Text2Onto

Task A (Text2Onto) involves the extraction of foundational ontological elements from raw unstructured text. It is divided into two subtasks: Term Extraction (A1), which identifies domain-specific vocabulary essential for populating ontologies, and Type Extraction (A2), which categorizes these terms into abstract classes, thus structuring knowledge representation for subsequent reasoning and semantic integration.

1.2 Task B: Term Typing

Task B focuses on assigning generalized semantic categories to previously extracted lexical terms. This task uses ontologies such as OBI (Biomedical Investigations), MatOnto (Materials Science), and SWEET (Earth and Environmental Science) to map terms accurately into their semantic classes, thereby structuring knowledge effectively and enabling enhanced reuse across diverse applications.

1.3 Task C: Taxonomy Discovery

In Task C, the goal is to discover hierarchical (is-a) relationships between pairs of types, essential for building structured taxonomic ontologies. This task spans multiple domains, leveraging specific ontologies like OBI (biomedical investigations), MatOnto (materials science), SWEET (environmental science), DOID (medical diseases), SchemaOrg (web knowledge), PROCO (chemical processes), FoodOn (food science), and PO (plant biology) to support robust hierarchical inference and knowledge management.

1.4 Task D: Non-Taxonomic Relation Extraction

Task D addresses the extraction of semantic relations beyond taxonomic hierarchies. It aims to identify meaningful associations like functional, compositional, and causal relationships, significantly enriching ontology utility. Subtasks involve identifying relations within specific ontologies such as SWEET (environmental and geoscience concepts), FoodOn (food ingredients and preparation methods), and GO (genomic relationships), addressing the previously identified challenge of LLMs in capturing deeper semantic nuances.

2. Literature Survey

[5] present domain-specific continual learning and prompt-tuning strategies for large language models (Llama-3-8B, GPT-3.5) in ontology learning tasks, demonstrating that knowledge-enriched training improves open-source model performance, though commercial models still outperform on benchmarks for term typing and taxonomy discovery.

[6] participated in the LLMs4OL 2024 Challenge, proposing prompt-based and classical machine learning techniques for ontology learning tasks, including term classification, taxonomy induction, and relation extraction. Leveraging LLMs such as GPT-4o and Llama-3, their methods achieved top-2 ranks in multiple subtasks, highlighting the promise of generative models for automated ontology construction.

[7] propose “semantic towers,” an extrinsic, hierarchical knowledge representation for ontology population and alignment. Through comprehensive experiments on WordNet and GeoNames, results demonstrate that, while intrinsic knowledge from LLMs achieves higher baseline accuracy, semantic towers improve semantic alignment and classification robustness, especially in low-resource and ambiguous scenarios.

[8] proposed a fine-tuned GPT-3.5 approach for term typing in ontology learning, evaluated in the LLMs4OL 2024 challenge across diverse datasets: WordNet, GeoNames, and UMLS. Their methodology involved dataset-specific prompt engineering and few-shot fine-tuning, yielding top leaderboard ranks in most cases. Results show LLMs can robustly identify and categorize ontology terms across domains, though challenges remain for highly ambiguous datasets such as GeoNames.

[9] introduce a soft prompt-tuning LLM framework for term typing in ontology learning, outperforming baselines on standard datasets but facing challenges in domains with complex class structures.

[10] address taxonomy discovery in ontologies by modeling parent-child extraction as a classification task. They compare fine-tuned BERT-Large and LLaMA 3 70B models, demonstrating that prompt quality and fine-tuning substantially impact performance. LLaMA slightly outperforms BERT, but both models achieve competitive results on the GeoNames dataset.

[11] present a hybrid method combining rule-based approaches with BERT models for ontology term typing and taxonomy extraction, demonstrating that the integration outperforms standalone large language models for LLMs4OL benchmark tasks.

[12] propose a RAG-based pipeline for automated ontology learning using LLMs, demonstrating promising results in term typing and relationship extraction, but highlighting limitations in specialized domains and the importance of model fine-tuning.

3. Methodology

3.1 Methodology for Task A: Text-to-Ontology (Text2Onto)

Our approach targets three specialized domains—*ecology*, *engineering*, and *scholarly*—and leverages the inferential capabilities of the large language model (LLM) `gemini-2.5-flash`. The overall pipeline is organized into three primary phases:

1. Corpus Preparation and Representation
2. Hierarchical Relation Extraction
3. LLM Inference and Knowledge Consolidation

3.1.1 Corpus Preparation and Representation

Domain-specific corpora were provided in JSON Lines (`.jsonl`) format, where each record contains a unique document identifier, title, and textual body. These corpora were uniformly processed across methods.

For **Method B**, we constructed a knowledge-enriched training set by explicitly associating each training document with validated term-type (ontology) mappings. This yielded a structured, searchable exemplar database that supports semantic retrieval in subsequent steps.

3.1.2 Hierarchical Relation Extraction Strategies

We developed two complementary strategies to extract term-type (hyponym-hypernym) relationships from text using the LLM, distinguished primarily by their approach to contextual guidance:

Method A: Heuristic-Guided Direct Extraction

This method applies a static, domain-agnostic prompt to extract term-type pairs directly from documents filtered via a keyword heuristic. Documents containing lexemes such as *type(s)*, *subtype(s)*, or their capitalized forms—common in definitional or taxonomic contexts—were selected for analysis.

For each candidate document, the prompt included:

- Clear instructions to identify and extract terms alongside their corresponding types.
- A one-shot example demonstrating the expected input-output JSON format.
- A strict constraint preventing the generation of any terms or types not present in the source text to minimize hallucination.
- **An explicit domain description (ecology, engineering, or scholarly) included in every prompt to contextualize the model's inference process.**

Method B: Retrieval-Augmented Extraction (RAE)

Adopting the Retrieval-Augmented Generation (RAG) paradigm, this method dynamically enriches prompts with highly relevant, domain-specific exemplars retrieved from the annotated training corpus.

The retrieval process involves:

1. **Text Vectorization:** Concatenating the title and body of each document, the entire corpus was vectorized using the **Term Frequency-Inverse Document Frequency (TF-IDF)** algorithm.
2. **Similarity Computation:** For each test document, cosine similarity was computed against all training document vectors.
3. **Best-Match Retrieval:** The highest scoring training document was selected as the exemplar for prompt augmentation.

The final prompt provided to the LLM consisted of:

- The full text and verified term-type pairs of the retrieved training exemplar.
- The full text of the target test document.
- Instructions to perform term-type extraction following the exemplar's format.
- **An explicit domain description clarifying the domain of the terms (ecology, engineering, or scholarly) to enhance contextual understanding.**

3.1.3 LLM Inference and Knowledge Consolidation

Both methods submitted their respective prompts to the `gemini-2.5-flash` model, configured to respond using the `application/json` MIME type. This structured output format ensures consistency and facilitates reliable parsing for evaluation.

The extracted term-type pairs were consolidated across documents and methods, enabling comprehensive evaluation of coverage, accuracy, and cross-domain generalization.

3.2 Methodology for Task B: Term Typing

The objective of this task was to assign semantic categories (e.g., *property*, *material*, *unit*) to a given list of technical terms from the material science domain. We designed a multi-stage hybrid approach combining deterministic lexical clustering with Large Language Model (LLM) based semantic disambiguation. The workflow consisted of three phases:

1. Data Preprocessing and Normalization
2. Lexical Clustering and Candidate Type Propagation
3. LLM-based Semantic Disambiguation and Final Classification

3.2.1 Lexical Clustering and Candidate Type Propagation

Terms were modeled as nodes in an undirected graph, where edges represented shared word tokens between terms. Connected components identified via Depth-First Search defined lexical clusters, computed separately for training and test sets.

For each test term, the best matching training cluster was selected based on maximal lexical overlap. The union of semantic types within this cluster formed a candidate type pool for that term. For test clusters, candidate pools of member terms were merged to form a final candidate type list. Terms lacking lexical matches defaulted to the full set of training types to maximize recall.

3.2.2 LLM-based Semantic Disambiguation and Final Classification

We employed the `gemini-2.5-pro` LLM to perform fine-grained semantic classification, selecting appropriate types from candidate pools.

The prompt engineered for the LLM included:

- Assignment of an expert persona knowledgeable in material science properties, materials, and units.
- Explicit scoping of domain knowledge relevant to the task.
- Constrained classification instructions requiring selection solely from the candidate type pool.
- Specification of structured JSON output mapping terms to their assigned types.
- **A dynamic domain description embedded in each prompt, providing contextual information about the relevant domain to guide and focus the model's reasoning process.**
- Few-shot examples illustrating the expected input-output format.

This multi-stage approach leverages lexical signals for recall and LLM semantic reasoning for precision, yielding robust term typing aligned with domain expertise. The model's structured JSON outputs were parsed to produce the final results.

3.3 Methodology for Task C: Taxonomy Discovery

The goal of this task was to automatically construct a taxonomic hierarchy, defined by parent-child (superclass-subclass) relationships, from a flat list of Schema.org types. We designed a hybrid multi-stage framework combining unsupervised clustering and Large Language Model (LLM)-based relation extraction.

3.3.1 Phase 1: Coarse-Grained Term Clustering

We explored two clustering strategies:

- **Lexical-Based Clustering:** Using a training set of known parent-child pairs, we constructed a lexical scaffold via a Union-Find graph algorithm connecting parents to children. Test terms were assigned to clusters by lexical overlap with these scaffolds or seeded new clusters when no overlap existed.
- **Semantic-Based Clustering (Preferred):** Terms were embedded into a high-dimensional vector space using domain-specific transformer models (e.g., SciBERT, BioBERT). The K-Means algorithm, with cluster count chosen by the Elbow Method [13] on Within-Cluster Sum of Squares (WCSS), partitioned terms into semantically coherent clusters.

3.3.2 Phase 2: Fine-Grained Relation Extraction via LLM

Each cluster was processed independently to improve precision. A carefully engineered prompt to `gemini-2.5-flash` instructed the model to:

- Recognize that terms belong to the Schema.org vocabulary and understand its hierarchical semantics.
- Extract explicit parent-child (superclass-subclass) relationships within the cluster.
- Follow structured JSON output formatting, listing pairs as objects with `parent` and `child` keys.

- Use provided illustrative examples to guide accurate relationship extraction.

3.3.3 Phase 3: Consolidation

The extracted parent-child pairs from all clusters were aggregated to form a complete taxonomic hierarchy. This final structure represents the inferred superclass-subclass organization of the original term list.

Note: The semantic clustering strategy is preferred due to its ability to capture conceptual similarity beyond surface lexical overlap, leading to more coherent clusters and improved hierarchical inference.

3.3.4 Methodology for ask D: Non-Taxonomic Relation Extraction

The task of non-taxonomic relationship extraction aims to identify and formalize complex connections (e.g., causal, part-whole, functional) between entities in a given domain. We developed two distinct, multi-stage methodologies to address this challenge, particularly in scenarios where the domain may not be explicitly defined beforehand. The primary approach, **Method A**, is a fully LLM-centric framework that performs sequential knowledge discovery, from domain inference to relation extraction. The secondary approach, **Method B**, is a hybrid framework that combines semantic embeddings and algorithmic clustering with subsequent LLM-based reasoning.

3.3.5 Method A: LLM-Centric Knowledge Discovery and Relation Extraction

This primary methodology leverages a series of targeted LLM prompts to progressively build context and extract relationships, simulating a human expert's reasoning process when confronted with a new domain. The process is divided into three sequential phases.

3.3.6 Phase 1: Automated Domain Inference

The process begins with the challenge of an uncontextualized set of entities. Given a list of terms and a list of potential relationship types, the first objective is to identify the underlying knowledge domain.

1. **Input**: A list of terms and a list of candidate relationship names.
2. **Process**: We prompt a large language model (`gemini-2.5-flash`) to act as a knowledge representation expert. The model is instructed to analyze the collective semantics of the terms and relations to deduce their common theme or scientific field.
3. **Output**: The model generates a structured JSON object containing a concise 'domain_name' (e.g., "Food Science and Production") and a detailed 'domain_description'. This description outlines the general area of study, the nature of the concepts, and the typical function of the relationships. This inferred context is critical, as it serves as the foundational knowledge base for all subsequent steps.

3.3.7 Phase 2: LLM-Inferred Semantic Clustering

With the domain now explicitly defined, the next phase aims to group the terms into semantically coherent clusters. This step reduces the problem space from an all-pairs comparison to a more focused, intra-cluster analysis.

1. **Input**: The list of terms, the list of candidate relationships, and the domain context (name and description) generated in Phase 1.
2. **Process**: A new prompt is sent to the LLM. This prompt provides the full domain context and instructs the model to act as an ontology reasoner. The LLM is tasked with building an implicit knowledge graph where terms are nodes. It evaluates every pair of terms, and if its internal knowledge of the identified domain suggests a valid connection exists via one of the specified relationship types, an undirected edge is formed. The model then identifies the connected components of this graph.
3. **Output**: The model returns a list of lists, where each inner list represents a cluster of semantically related terms. Terms with no inferred connections form their own singleton clusters.

3.3.8 Phase 3: Intra-Cluster Relation Triplet Generation

This final phase performs the high-precision task of explicitly defining the relationships within the semantically-related groups identified in the previous step.

1. **Input**: Each individual term cluster and the list of candidate relationships, again contextualized by the domain description.
2. **Process**: To ensure the highest degree of reasoning, we employ a more powerful model (`gemini-2.5-pro`) for this task. For each cluster, a new prompt instructs the model to examine all possible pairs of terms *within that cluster*. Using its domain-specific knowledge, it generates all valid relationship triplets ('head', 'relation', 'tail') that can be formed using the provided list of relationships.
3. **Output**: The model returns a list of relationship triplets for each cluster. These are then aggregated into a single, comprehensive list of all non-taxonomic relationships discovered in the corpus.

3.4 Method B: Hybrid Semantic Clustering and Relation Extraction

This secondary methodology provides an alternative path for the initial clustering phase, replacing LLM-inferred grouping with a combination of pre-trained embedding models and unsupervised machine learning algorithms.

3.4.1 Phase 1: Semantic Vectorization

This phase converts the list of terms into a numerical representation that captures their semantic meaning.

1. **Input**: A list of terms.
2. **Process**: We utilize a domain-specific, pre-trained transformer model (e.g., `BioBERT` for biological terms, `RecipeBERT` for food-related terms) to generate a high-dimensional vector embedding for each term. This process maps terms with similar meanings to points that are close to each other in vector space.

3.4.2 Phase 2: Unsupervised Algorithmic Clustering

With the terms represented as vectors, we apply a standard clustering algorithm to group them based on semantic proximity.

1. **Input**: The term embeddings generated in the previous step.
2. **Process**: We employ the **K-Means** clustering algorithm. To determine the optimal number of clusters (k), we use the **Elbow Method**, which analyzes the Within-

Cluster Sum of Squares (WCSS) across a range of k values to find the point of diminishing returns.
3. **Output**: The algorithm partitions the terms into k distinct clusters based on the similarity of their semantic embeddings.

3.4.3 Phase 3: LLM-based Intra-Cluster Relation Identification

This final step is analogous to Phase 3 of Method A. Having obtained clusters through algorithmic means, we now use an LLM for the final, high-precision extraction of relationship triplets within each cluster, following the same prompting and reasoning strategy described in Section 1.1.3.

4. Results and Analysis

We evaluated our clustering-driven, LLM-enhanced framework across four major ontology learning tasks defined in the LLMs4OL 2025 challenge. Table 1 summarizes the F1-scores and rankings achieved in each subtask.

Table 1. *F1-scores and Rankings for Each Sub-task in LLMs4OL 2025*

Task	Sub-task	F1-score	Rank
Term Extraction	A1.2 - Scholarly	0.4578	4
Term Extraction	A1.3 - Engineering	0.4302	6
Type Extraction	A2.1 - Ecology	0.5535	4
Type Extraction	A2.2 - Scholarly	0.2500	7
Type Extraction	A2.3 - Engineering	0.2545	5
Term Typing	B1 - OBI	0.8021	5
Term Typing	B2 - MatOnto	0.4872	5
Term Typing	B3 - SWEET	0.3297	7
Taxonomy Discovery	C1 - OBI	0.2273	3
Taxonomy Discovery	C2 - MatOnto	0.4473	4
Taxonomy Discovery	C5 - SchemaOrg	0.2609	4
Taxonomy Discovery	C6 - PROCO	0.2601	1
Taxonomy Discovery	C8 - PO	0.2106	3
Taxonomy Discovery	C10 - Blind	0.5735	1
Taxonomy Discovery	C11 - Blind	0.4684	1
Non-Taxonomic Relation Extraction	D1 - SWEET	0.6263	1
Non-Taxonomic Relation Extraction	D2 - FoodOn	0.0084	1
Non-Taxonomic Relation Extraction	D4 - Blind	0.5051	1

Our model consistently ranked among the top performers, particularly in subtasks requiring generalization across unseen domains (e.g., C6–C11 and D1–D4). Notably, our semantic clustering coupled with transformer-based embeddings (e.g., BioBERT, MaterialsBERT) enabled accurate identification of taxonomic and non-taxonomic relationships. The strong performance in blind subtasks (C10, C11, D4) underscores the adaptability of our LLM-inference framework to unknown domains.

However, subtasks involving domain-specific jargon (e.g., A2.2 – Scholarly and B3 – SWEET) exhibited lower F1-scores, revealing limitations in contextual understanding and type assignment precision, even with clustering assistance. These results affirm that clustering offers structural benefits, but domain adaptation remains essential for enhanced type inference.

Overall, our system demonstrated robustness and competitive performance across varied ontology learning tasks, validating the efficacy of combining clustering with domain-aware prompting and transformer-based semantic models.

5. Conclusion

In this paper, we presented a clustering-based ontology learning framework built atop domain-specialized large language models to address the four primary tasks of the LLMs4OL 2025 challenge. Our method integrates lexical structure with semantic embeddings and prompt-based inference, enabling scalable ontology construction across tasks such as term and type extraction, term typing, taxonomy discovery, and relation modeling.

The experimental results confirm that clustering-driven representations, when coupled with transformer-based LLMs, can yield competitive performance across subtasks and domains. We observed significant success in blind and generalization-focused subtasks, where structured clustering improved model reasoning and reduced noise in domain inference. However, performance in linguistically complex domains—like scholarly or SWEET ontologies—indicates a need for deeper domain alignment and robust context modeling.

Future work will explore fine-tuning with curriculum learning, enhancing prompt personalization, and integrating symbolic reasoning or external knowledge graphs to address limitations in semantic disambiguation. Overall, this study reinforces the promise of combining unsupervised structuring with LLM-based reasoning for automated and domain-adaptive ontology learning.

Data availability statement

The task organizers provided the data used in this study as part of the "LLMs4OL 2025 Overview: The 2nd Large Language Models for Ontology Learning Challenge" [1]. Access to the data is subject to the terms and conditions specified by the organizers.

Author Contributions

Pankaj Kumar Goyal: Data curation, Methodology, Validation, Implementation., Writing – Original Draft
Sumit Singh: Conceptualisation, Writing – Original Draft, Writing – Review & Editing, Investigation.
Uma Shanker Tiwary: Supervision.

Competing interests

The authors declare that they have no competing interests.

References

[1] H. Babaei Giglou, J. D'Souza, N. Mihindukulasooriya, and S. Auer, "Llms4ol 2025 overview: The 2nd large language models for ontology learning challenge", *Open Conference Proceedings*, 2025.
[2] P. Shetty et al., "A general-purpose material property data extraction pipeline from large polymer corpora using natural language processing", *npj Computational Materials*, vol. 9,

no. 1, p. 52, 2023, Implements and releases MaterialsBERT as 'pranav-s/MaterialsBERT' on Hugging Face. DOI: 10.1038/s41524-023-00994-5.

[3] J. Lee et al., "Biobert: A pre-trained biomedical language representation model for biomedical text mining", *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.

[4] *Chat & ask ai: Your advanced ai chatbot*, https://askaichat.app/chat, Accessed July 2025.

[5] Y. Peng, Y. Mou, B. Zhu, S. Sowe, and S. Decker, "Rwth-dbis at llms4ol 2024 tasks a and b: Knowledge-enhanced domain-specific continual learning and prompt-tuning of large language models for ontology learning", *Open Conference Proceedings*, vol. 4, pp. 49–63, Oct. 2024. DOI: 10.52825/ocp.v4i.2491. [Online]. Available: https://www.tib-op.org/ojs/index.php/ocp/article/view/2491.

[6] P. Kumar Goyal, S. Singh, and U. Shanker Tiwary, "Silp_nlp at llms4ol 2024 tasks a, b, and c: Ontology learning through prompts with llms", *Open Conference Proceedings*, vol. 4, pp. 31–38, Oct. 2024. DOI: 10.52825/ocp.v4i.2485. [Online]. Available: https://www.tib-op.org/ojs/index.php/ocp/article/view/2485.

[7] H. Abi Akl, "Dsti at llms4ol 2024 task a: Intrinsic versus extrinsic knowledge for type classification: Applications on wordnet and geonames datasets", *Open Conference Proceedings*, vol. 4, pp. 93–101, Oct. 2024. DOI: 10.52825/ocp.v4i.2492. [Online]. Available: https://www.tib-op.org/ojs/index.php/ocp/article/view/2492.

[8] A. Barua, S. Saki Norouzi, and P. Hitzler, "Daselab at llms4ol 2024 task a: Towards term typing in ontology learning", *Open Conference Proceedings*, vol. 4, pp. 77–84, Oct. 2024. DOI: 10.52825/ocp.v4i.2489. [Online]. Available: https://www.tib-op.org/ojs/index.php/ocp/article/view/2489.

[9] T. Phuttaamart, N. Kertkeidkachorn, and A. Trongratsameethong, "The ghost at llms4ol 2024 task a: Prompt-tuning-based large language models for term typing", *Open Conference Proceedings*, vol. 4, pp. 85–91, Oct. 2024. DOI: 10.52825/ocp.v4i.2486. [Online]. Available: https://www.tib-op.org/ojs/index.php/ocp/article/view/2486.

[10] S. M. H. Hashemi, M. Karimi Manesh, and M. Shamsfard, "Skh-nlp at llms4ol 2024 task b: Taxonomy discovery in ontologies using bert and llama 3", *Open Conference Proceedings*, vol. 4, pp. 103–111, Oct. 2024. DOI: 10.52825/ocp.v4i.2483. [Online]. Available: https://www.tib-op.org/ojs/index.php/ocp/article/view/2483.

[11] C. A. Atezong Ymele and A. Jiomekong, "Tsotsalearning at llms4ol tasks a and b : Combining rules to large language model for ontology learning", *Open Conference Proceedings*, vol. 4, pp. 65–76, Oct. 2024. DOI: 10.52825/ocp.v4i.2484. [Online]. Available: https://www.tib-op.org/ojs/index.php/ocp/article/view/2484.

[12] M. Sanaei, F. Azizi, and H. Babaei Giglou, "Phoenixes at llms4ol 2024 tasks a, b, and c: Retrieval augmented generation for ontology learning", *Open Conference Proceedings*, vol. 4, pp. 39–47, Oct. 2024. DOI: 10.52825/ocp.v4i.2482. [Online]. Available: https://www.tib-op.org/ojs/index.php/ocp/article/view/2482.

[13] E. Umargono, J. E. Suseno, and S. V. Gunawan, "K-means clustering optimization using the elbow method and early centroid determination based on mean and median formula", in *Proceedings of the 2nd International Seminar on Science and Technology (ISSTEC 2019)*, Atlantis Press, 2020, pp. 121–129, ISBN: 978-94-6239-168-0. DOI: 10.2991/assehr.k.201010.019. [Online]. Available: https://doi.org/10.2991/assehr.k.201010.019.